UC Berkeley
Dissertations

Title
Self-Organizing and Optimal Control for Nonlinear Systems

Permalink
https://escholarship.org/uc/item/1hd7c0dv

Author
Dong, Wenjie

Publication Date
2009-09-01

eScholarship.org Powered by the California Digital Library
University of California

University of California Transportation Center
UCTC Dissertation No. 152

Self-Organizing and Optimal Control for Nonlinear Systems

Wenjie Dong
University of California, Riverside
June 2009

UNIVERSITY OF CALIFORNIA
RIVERSIDE

Self-Organizing and Optimal Control for Nonlinear Systems

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Electrical Engineering

by

Wenjie Dong

June 2009

Dissertation Committee:

Professor Jay A. Farrell, Chairperson
Professor Jie Chen
Professor Yingbo Hua

Copyright by
Wenjie Dong
2009

The Dissertation of Wenjie Dong is approved:

Committee Chairperson

University of California, Riverside

Acknowledgments

I would like to take this opportunity to express my gratitude to my advisor Prof. Jay A. Farrell for his excellent guidance throughout the course of this dissertation and for his support during my graduate studies at the University of California, Riverside. I truly consider it a great privilege to have the opportunity to work with him.

I would like to thank my committee members, Prof. Jie Chen and Prof. Yingbo Hua, for serving on my committee and contributing towards enhancing the quality of my dissertation in several ways.

I also thank all my lab members in the Robotics and Control Laboratory for their help and discussions.

I profoundly thank my wife Lingyun. Her constant understanding and support is the source of all my strength and courage.

This material is based upon work supported by the National Science Foundation (NSF) under Grant No. 0701791 and the Dissertation Grant from the University of California Transportation Center (UCTC). I would like to thank the NSF and UCTC for the financial support. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation

To my parents.

ABSTRACT OF THE DISSERTATION

Self-Organizing and Optimal Control for Nonlinear Systems

by

Wenjie Dong

Doctor of Philosophy, Graduate Program in Electrical Engineering
University of California, Riverside, June 2009
Professor Jay A. Farrell, Chairperson

Vehicle formation control is one of important research topics in transportation. Control of uncertain nonlinear systems is one of fundamental problems in vehicle control. In this dissertation, we consider this fundamental control problem. Specially, we considered self-organizing based tracking control of uncertain nonaffine systems and optimal control of uncertain nonlinear systems. In tracking control of nonaffine systems, a self-organizing on-line approximation based controller is proposed to achieve a prespecified tracking accuracy, without using high-gain control nor large magnitude switching. For optimal control of uncertain nonlinear systems, we considered point-wise min-norm optimal control of uncertain nonlinear systems and approximately optimal control of uncertain nonlinear systems. In point-wise non-norm optimal control, optimal regulation and optimal tracking controllers were proposed with the aid of locally weighted learning observers. By introducing control Lyapunov functions and redefining the optimal criterions, analytic controllers were proposed and were optimal in the sense of min-norm. In approximately optimal control of uncertain nonlinear systems, adaptive optimal controllers were proposed with the aid of

iterative approximation techniques and adaptive control. By iteratively learning, the difficulty of solving Hamilton-Jacobian-Bellman (HJB) equation is overcome. The proposed adaptive optimal algorithms can be applied to solve optimal control problem of a large class of nonlinear systems. To show effectiveness of the proposed controllers for above problems, simulations were done in computers.

Contents

List of Figures

Chapter 1

Introduction

This thesis consider self-organizing and optimal control of nonlinear systems that have model error significant enough to affect the achievable performance of the closed-loop system. As motivation, one example is presented in Section 1.1.

1.1 Uncertainty in Vehicle Formation Control

With the development of economy and population growth, an increasing number of vehicles are on highways. Transportation issues such as roadway capacity, traffic congestion, and highway safety have arisen in the past decades. One method to deal with these issues is to develop Intelligent Vehicle Highway Systems (IVHS). As an enabling technology, increasing attention has been paid to the development of Advanced Vehicle Control Systems (AVCs) to increase lane utilization without sacrificing safety requirements. For example, AVCs can be designed such that vehicles automatically travel in closely spaced platoons proposed [65]. It has been estimated that the close formation of vehicles could increase highway capacity by a factor or two or three with automatic control [66]. To study the stability of vehicle platoons, the "string stability" problem has been studied as early as 1977 [11]. String-stability ensures that range errors do not increase as they propagate along the vehicle stream.

Vehicle formation stability is a more general concept than the string stability and includes string stability as a special case. In recent years, formation control has been an active research area. Several methods for formation control have been proposed. In [2, 5, 31, 36], behavior-based control methods were proposed. In [7, 58], virtual structure based method was proposed by implementing decentralized trajectory-following controllers on each vehicle. In [16, 72], leader-follower approach was proposed by designating one agent as a leader and the others as followers. In [37, 46], artificial potentials were applied to solve the cooperative control problem with the aid of other techniques, such as graph theory, virtual bodies/leaders, etc. In [23, 29, 34, 40, 43, 47, 59, 71], results from graph theory were applied to cooperative control of multiple systems. The author has studied the vehicle formation based on kinematics of each vehicle. Decentralized feedback control laws were proposed for a group of vehicles such that they converge to a desired stationary formation or a desired formation which moves along a desired trajectory with the aid of results from graph theory [17–20]. However, all vehicles have dynamics. It is important to study the control problem of vehicles with dynamics. For a vehicle, its dynamics are nonlinear and are hard to be obtained due to the complicated structure of vehicles and varying loads for different missions. In this thesis, we consider several fundamental control problems applicable to uncertain nonlinear systems. Solutions to these fundamental problems have extensive applications in vehicle formation control applications and other practical applications such as color printing systems, autonomous vehicles, etc.

For nonlinear systems [32, 33, 41, 63, 67], there are two main categories of methods to address model error: (1) domination of the error with high gain and switching; and, (2) adaptation or learning. Methods such as Lyapunov redesign and sliding mode control are applicable when a known bound on the model uncertainty is available. These methods can retain convergence of solutions to a desired trajectory. Nonlinear damping methods guarantee boundedness of solutions and are applicable when a priori bounds on the model uncertainty are not known. Backstepping allows such methods to be extended to some applications where the model uncertainty does not satisfy a matching condition. When model error bounds are not known at the design stage, such bounds can be estimated on-line [30, 52]. Lyapunov redesign, sliding mode, and nonlinear damping use (potentially) large magnitude, high frequency switching of the control signal to dominate the model error. Successful implementation of such controllers requires high bandwidth actuators and uses significant control energy. In addition, high-frequency switching can excite high-frequency unmodeled effects. Those methods may not result in comfortable rides if applied to passenger vehicles. Alternatively, adaptive and learning based methods address the model error by learning either the model error or the control law during the system operation. To minimize the required on-line learning, such methods should capitalize on the known design model to the greatest extent possible. In particular, when the model error has a parametric form (i.e., $\theta^\top \phi(x)$ with θ unknown and $\phi(x)$ of a known form determined by the physics of the problem) then standard nonlinear adaptive methods are applicable [33, 67]. When the model error does not have a known parametric form, then more general on-line function approximation-based techniques are required, e.g., [22, 26].

In this thesis, we will consider several control problems related to uncertain nonlinear systems with the aid of on-line learning techniques. Solutions to these control problems have many applications in vehicle control. These problems and their current states are discussed in the following several sections.

1.2 Self-organizing Online Learning

In the past decades, on-line approximation based control has been considered extensively in e.g., [4, 9, 10, 12, 13, 21, 25, 30, 35, 38, 39, 49–51, 60, 76, 77]. The design and analysis of adaptive controllers involving on-line approximation to achieve stability and accurate trajectory tracking in the presence of unknown or partially unknown nonlinear dynamics have been well developed.

In general, on-line approximation based controllers cannot achieve an exact modeling of unknown nonlinearities, *inherent approximation errors* could arise even if optimal approximator parameters were selected. Under reasonable assumption on the basis function and the function to be approximated, for any given $\epsilon > 0$, if the network approximator has a sufficiently large number of nodes, then ϵ approximation accuracy can be achieved by proper selection of the approximator parameters [27, 48]. Thus, to meet ϵ approximation accuracy, one approach is to allocate a sufficient large number of learning parameters. However, allocating too many learning parameters bears the danger of over-parameterizing the approximation. This may have computation and performance penalties. Another approach is to define the approximator structure automatically during operation. With these motivations, nonlinear adaptive control with function approximation employing automatic structure adaptation has been discussed in a few articles [4, 10, 13, 21, 44, 45, 60, 64, 76, 77]. Articles [10, 60] used wavelet networks and adapted the structure of the network in response to the evaluation of the magnitude of the output weights by "hard-thresholding". Smoothly interpolated linear models were considered in [13].

In [4, 64], local approximators within localized receptive field were defined and the on-line approximation was tuned in a local region without affecting the approximation accuracy previously achieved in other regions. Therefore, the function approximation structure is able to retain approximation accuracy as a function of the operating point. As in [13], the structure adaptation is based on exploration: if none of the existing basis functions is excited, then a new node is allocated. These articles also use gradient descent to adjust the distance metric of each local approximator so that each receptive field is tuned according to the local curvature properties of the unknown function. In [13, 44, 45], linearly parameterized local models were used, which is a special case of the Receptive Field Weighted Regression (RFWR) approach. No stability results are given in [4, 64]. The common shortcoming of the approaches in [10, 13, 44, 45, 60] is that (i) they only address the stability analysis for the state and the approximator parameters, not the change in the number of basis functions, and (ii) the structure adaptation algorithms are defined by the trajectory, not by the performance. New approximator nodes are added when the current state is sufficiently far from all existing receptive field centers, whether or not additional approximation accuracy is required. Recent articles [21, 77] developed an approach where the approximation structure is adjusted during system operation, based on the observed trajectory tracking performance. A self-organized state estimation approach is developed in [76] and organizes the approximator structure based on estimation error. Article [21, 76, 77] focus on models that are affine in the control variable.

In Chapter 2, we consider the tracking control of a n-th order nonaffine system. The goal is to design a self-organizing on-line approximation based controller to achieve a prespecified tracking accuracy, without using high-gain control nor large magnitude switching. For non-affine system control, article [75] proposed a high gain controller.

1.3 Point-wise Min-norm Optimal Control of Uncertain Nonlinear Systems

Optimal control theory was formally developed about fifty years ago in the seminal works of L. S. Pontryagin [53] in the former Soviet Union and R. Bellman [8] in the United States. While Pontryagin introduced the minimum principle, which gave necessary conditions for the existence of optimal trajectories, Bellman introduced the concept of dynamic programming. The development of dynamic programming led to the notion of the celebrated Hamilton-Jacobi Bellman (HJB) partial differential equation, which had the value function as its solution. For the Linear Quadratic Gaussian (LQG) problem [3], i.e., the $\mathcal{H}_2$ optimal control problem, the HJB partial differential equation becomes two separate Riccati equations, which could be solved very efficiently by several algorithms. For optimal control of general nonlinear systems, it is hard to obtain the optimal controllers. One reason is that the HJB equation is extremely hard to solve for nonlinear systems. Also, there is no efficient numeric algorithm to solve it.

For the optimal control of uncertain nonlinear systems, one approach is to learn the unknown system model offline and then design optimal controllers based on the estimate models. Another approach is to apply nonlinear $\mathcal{H}_\infty$ control theory and the optimal controllers are obtained based on the Hamilton-Jacobi-Isaac (HJI) equations [15, 28], which are hard to solve. Neural network based controllers were proposed in [1] to optimize both $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms of performances for uncertain nonlinear systems.

In Chapter 3, we consider optimal regulation and optimal tracking control of uncertain nonlinear systems. To deal with the uncertain terms, we first propose a locally weighted learning observer (LWLO) to estimate the unknown nonlinear system. Based on the approximators that result from locally weighted learning observer, point-wise min-norm problems are defined. For the defined optimal problems, analytic controllers are proposed based on selected Lyapunov functions.

1.4 Approximately Optimal Control of Uncertain Nonlinear Systems

For the methods derived in Chapter 3, the proposed controllers are optimal in the sense of point-wise min-norm of the control input subject to a constraint on the state. To solve the optimal control of nonlinear systems with respect to a cost function of the state and the input, many techniques have been developed. For example, receding horizon control [42], approximate dynamic programming [54], approximating the HJB equation, approximation of value functions [6, 62], etc.

For approximate dynamic programming, there are many methods [54]. For example, value iteration, policy iteration, Q-learning, value function approximation, adaptive critic design, etc. Adaptive critic design (ACD) is derived from approximate dynamic programming [57]. Adaptive critic designs are based on an algorithm that cycles between a policy improvement routine and a value-determination operation. At each optimizing cycle, the algorithm approximates the optimal control law and the value function based on the state. In ACD, the optimal control problem is solved without backforward iteration in time.

For continuous nonlinear systems, approximately optimal controllers for regulation were proposed for continuous dynamic systems in [62]. By the proposed method, a series of controllers were proposed based on a series of value functions. It was shown that the series of value functions decreased and converged to the minimal value function corresponding to the optimal control. The series of controllers derived from the series of value functions converged to the optimal controller. In [6], in order to solve the optimal regulation problem the Galerkin approximation method was applied to approximate the Hamilton-Jacobi-Bellman (HJB) equation over a compact set containing the origin. It was shown that the Galerkin approximation converged to the solution of the generalized Hamilton-Jacobi-Bellman (GHJB) equation and that the resulting approximate control was stabilizing on the same region as the initial control under some assumptions. However, both controller design methods are offline and require exact knowledge of the system dynamics. To make the design procedure online, an adaptive optimal controller was proposed for regulation in [73]. The proposed optimal controller converged to the optimal controller. In this design procedure, the value functions were approximated by a neural network. In order to solve the weight parameters a non-singularity condition was required at each step.

In Chapter 4, we consider the optimal control of the uncertain nonlinear system shown in (4.1). By combining the results in adaptive critic design and the approximately optimal control algorithms in [6, 62] we propose adaptive approximately optimal control algorithms for optimal regulation.

1.5 Contributions of This Dissertation

The results in this dissertation make the following contributions.

- **Self-organizing online learning:** Tracking control of an n-th order nonaffine system was considered. Self-organizing on-line approximation based controller was proposed to achieve a prespecified tracking accuracy, without using high-gain control nor large magnitude switching. It was shown that our proposed controllers can achieve the prespecified tracking performance and require less computation during control. Compared with the existing results, the contribution of this part is that the tracking control of high-order nonaffine systems is solved with the aid of a self-organizing approximation and a low gain computationally efficient controller.

- **Optimal control of uncertain nonlinear systems:** Optimal regulation and optimal tracking control of uncertain nonlinear systems were considered. Locally weighted learning observers (LWLOs) were proposed to estimate the unknown nonlinear systems. Based on the approximators that result from locally weighted learning observers, point-wise min-norm problems were defined. For the defined optimal problems, analytic controllers were proposed based on selected Lyapunov functions. The contribution of this part is that the optimal regulation and the optimal tracking control of uncertain nonlinear systems were solved by integrating local learning algorithms to point-wise min-norm controllers.

- **Approximately optimal control of uncertain nonlinear systems:** Optimal control of the uncertain nonlinear systems was considered. By combining the results in adaptive critic design and the approximately optimal control algorithms in [62, 73, 74] adaptive approximately optimal control algorithms were proposed. In the proposed algorithms the controllers are updated according to the information of the value functions and converge to the optimal controllers. Compared with the offline optimal controller design algorithms in [6, 62], the proposed algorithms work online and for unknown systems.

Chapter 2

Tracking Control of Nonaffine Systems: A Self-organizing Approximation Approach

In this chapter we consider the tracking control of a n-th order nonaffine system. Our goal is to design a self-organizing on-line approximation based controller to achieve a prespecified tracking accuracy, without using high-gain control nor large magnitude switching. Towards this end, in the operational region we propose an adaptive controller based on the self-organizing idea in [77] and approximation ideas in [25, 49]. To make sure the state of the system comes into the operational region a sliding mode control is proposed as a supervisory control. Once the state of the system comes into the operational region the adaptive controller takes over the control. We will show that our proposed controllers will achieve the prespecified tracking performance and require less computation during control.

The remainder of this chapter is organized as follows. In Section 2.1, the problem considered in this article is defined. In Section 2.2, the basic controller structure is presented. In Section 2.3, a local weighted learning algorithm and a structure adaption are proposed. Section 2.4 proves the stability of the proposed controller. Section 2.5 shows the effectiveness of the proposed controller by a numerical example. The last section concludes this chapter.

2.1 Problem Statement

Consider single-input single-output (SISO), input-state feedback linearizable systems of the form

$$\dot{x}_i = x_{i+1}, \qquad 1 \leq i \leq n-1 \tag{2.1}$$

$$\dot{x}_n = h(x, u) \tag{2.2}$$

$$y = x_1 \tag{2.3}$$

where $x = [x_1, \cdots, x_n]^\top \in R^n$ is the state vector, $y \in R$ is the output, and $u \in R$ is the control signal. The function $h(x, u)$ represents nonlinear effects that are unknown at the design stage. The function h is assumed to be differentiable with respect to u.

Given a desired bounded trajectory $x_d(t)$ with derivatives $x_d^{(i)}(t),\ i = 1, \ldots, n$, each of which is available and bounded $\forall t \geq 0$. For convenience, we denote

$$x_c(t) = [x_{1c}, x_{2c}, \ldots, x_{nc}]^\top = [x_d, x_d^{(1)}, \ldots, x_d^{(n-1)}]^\top.$$

Control Problem: Design the control signal u to steer $x_1(t)$ to track the desired trajectory $x_d(t)$ and to achieve boundedness for the states x_i for $i = 2, \ldots, n$.

Assumption 2.1.1 *For any $x \in \mathcal{R}^n$ and $u \in \mathcal{R}$,*

$$\epsilon_0(x) < \frac{\partial h(x,u)}{\partial u} < 2c(x) \tag{2.4}$$

$$|h(x,0)| \leq b(x) \tag{2.5}$$

where $\epsilon_0(x)$, $c(x)$, and $b(x)$ are known positive functions.

Remark 2.1.2 *In Assumption 2.1.1, the first inequality in (2.4) makes sure that the system is controllable at any time. The second inequality in (2.4) will ensure that $f(x)$ defined in eqn. (2.16) is unique and continuous (see Lemma 2.2.3). The bound $b(x)$ in (2.5) will be used by the supervisory controllers to ensure that the tracking error comes into and stays in a bounded operational region which is chosen by the designer.*

2.2 Tracking Errors and Basic Control Structure

Throughout the article the tracking error components are defined as

$$\tilde{x}_i = x_i - x_{ic}, \quad 1 \leq i \leq n$$

where $\tilde{x}$ is the tracking error vector defined as

$$\tilde{x} = x - x_c = [\tilde{x}_1, \ldots, \tilde{x}_n]^\top.$$

Note that

$$\dot{\tilde{x}}_i = \tilde{x}_{i+1}, \quad 1 \leq i \leq n-1.$$

Following the derivation in [67], let

$$e(t) = L^\top \tilde{x}(t) \tag{2.6}$$

where $L = [l_1, l_2, \ldots, l_{n-1}, 1]^\top$ is a constant vector. Note that $L^\top \tilde{x} = 0$ defines an $(n-1)$-dimensional hyperplane in $\Re^n$. The absolute value of $e(t)$ represents the distance of $\tilde{x}(t)$ from this hyperplane. On the hyperplane $e(t) = 0$, the dynamics of $\tilde{x}_1$ are defined by

$$\begin{aligned}
\tilde{x}_n + l_{n-1}\tilde{x}_{n-1} + \ldots + l_3\tilde{x}_3 + l_2\tilde{x}_2 + l_1\tilde{x}_1 &= 0 \\
\dot{\tilde{x}}_{n-1} + l_{n-1}\dot{\tilde{x}}_{n-2} + \ldots + l_3\dot{\tilde{x}}_2 + l_2\dot{\tilde{x}}_1 + l_1\tilde{x}_1 &= 0 \\
&\vdots \\
(s^{n-1} + l_{n-1}s^{n-2} + \cdots + l_3 s^2 + l_2 s + l_1)\tilde{x}_1 &= 0
\end{aligned}$$

where s is the Laplace variable; therefore, L is selected so that

$$(s^{n-1} + l_{n-1}s^{n-2} + \cdots + l_3 s^2 + l_2 s + l_1) = 0$$

is a Hurwitz polynomial. In this case, the transfer function

$$\frac{\tilde{x}_1(s)}{e(s)} = \frac{1}{s^{n-1} + l_{n-1}s^{n-2} + \cdots + l_3 s^2 + l_2 s + l_1}$$

is Bounded-Input-Bounded-Output (BIBO) stable. If $e(t)$ can be shown to be bounded for all $t \geq 0$, then each $\tilde{x}_i, 1 \leq i \leq n$ is bounded.

To allow the bounds to be easily expressed, we choose L such that

$$\begin{aligned}
e(t) &= \left(\frac{d}{dt} + \lambda\right)^{n-1} \tilde{x}_1 = \sum_{i=1}^{n} C_{n-1}^{i-1} \left[\frac{d^{i-1}}{dt^{i-1}}\tilde{x}_1\right] \lambda^{n-i} \\
&= [\lambda^{n-1}, C_{n-1}^{1}\lambda^{n-2}, \ldots, C_{n-1}^{n-2}\lambda, 1]\tilde{x}
\end{aligned}$$

for some constant $\lambda > 0$. This implies that the vector L in (2.6) is defined as

$$l_i = C_{n-1}^{i-1}\lambda^{n-i}, \quad 1 \leq i \leq n$$

where

$$C_{n-1}^{i-1} = \frac{(n-1)!}{(n-i)!(i-1)!}$$

is the binomial coefficient. The transfer functions to $\tilde{x}_i$ from e are

$$\begin{aligned}\frac{\tilde{x}_i(s)}{e(s)} &= \frac{\tilde{x}_1(s)}{e(s)} s^{i-1} = \frac{s^{i-1}}{(s+\lambda)^{n-1}} \\ &= \frac{1}{(s+\lambda)^{n-i}} \cdot \left(1 - \frac{\lambda}{s+\lambda}\right)^{i-1}, \ 1 \leq i \leq n.\end{aligned}$$

The advantage of defining $e(t)$ in this manner is that if there exists a constant $\mu_e > 0$ such that the magnitude of e is bounded as $|e(t)| \leq \mu_e, \ \forall t \geq 0$, then the tracking errors are asymptotically bounded by

$$|\tilde{x}_i(t)| \leq 2^{i-1} \lambda^{i-n} \mu_e, \ 1 \leq i \leq n, \tag{2.7}$$

which yields

$$\|\tilde{x}(t)\|_2 \quad \leq \quad \|\lambda_v\|_2 \mu_e \text{ as } t \to \infty \tag{2.8}$$

with

$$\lambda_v^\top = [\lambda^{1-n}, 2\lambda^{2-n}, \ldots, 2^{n-2}\lambda, 2^{n-1}]$$

and $\|\cdot\|_2$ being the 2-norm of a vector. See page 279-280 of [67] for additional detail.

The self-organizing on-line approximation based controller developed in the subsequent sections is designed to maintain stability and to achieve a tracking accuracy of $|e(t)| < \mu_e$ with μ_e prespecified at the design stage. If L is selected as in the previous paragraph, then $|e(t)| < \mu_e$ ensures that

$$|\tilde{x}_1| < \frac{1}{\lambda^{n-1}} \mu_e \doteq \mu_x$$

as $t \to \infty$. Given a positive constant $\sigma > 0$, let

$$\mathcal{D}^n = \{x \in R^n |\ |\tilde{x}_i| \le 2^{i-1}\lambda^{i-n}\sigma,\ 1 \le i \le n\}.$$

It is obvious that $\mathcal{D}^n$ is bounded since x_c and σ are bounded. Noting (2.7), if $|e| \le \sigma$, then $x \in \mathcal{D}^n$. We call the region $\mathcal{D}^n$ *the operational region.* It can be adjusted by the choice of σ.

With the definition of e in (2.6),

$$\dot{e} = \Lambda + h(x,u) = \Lambda + cu + (h(x,u) - cu) \tag{2.9}$$

where c is the bounded function in eqn. (2.4), and

$$\Lambda = \lambda^{n-1}\tilde{x}_2 + C_{n-1}^1\lambda^{n-2}\tilde{x}_3 + \cdots + C_{n-1}^{n-2}\lambda\tilde{x}_n - x_d^{(n)}. \tag{2.10}$$

We choose the control law

$$u = \begin{cases} \frac{1}{c}\left(-Ke - \Lambda - u_{ad} - \epsilon_f \text{sat}\left(\frac{e}{\mu_e}\right)\right), & |e| \le \sigma \\ -\frac{1}{\epsilon_0}[Ke + (|\Lambda| + b(x))\text{sign}(e)], & |e| > \sigma \end{cases} \tag{2.11}$$

where constant $K > 0$, $\sigma(> \mu_e)$ is a positive constant, $\epsilon_f(> 0)$ will be defined later,

$$\text{sat}(y) = \begin{cases} \text{sign}(y), & \text{if } |y| > 1 \\ y, & \text{otherwise} \end{cases}$$

and u_{ad} is defined later in (2.28).

Remark 2.2.1 *If $|e| > \sigma$, the controller is a sliding mode control [61, 67]. If $|e| \le \sigma$, the controller is self-organizing in a manner that will be defined in the following section. We choose the control law (2.11) to be a switching controller because we want to use a high gain* *control to steer the state of the system to the bounded operational region $\mathcal{D}^n$. Within this operational region an adaptive learning controller is in charge of the control. Hence, in the operational region the control does not use large magnitude high gain switching even though the system model is unknown (see Fig. 2.1 and Remark 2.4.2 for more details).*

Lemma 2.2.2 *If $|e| > \sigma$, with the control in (2.11), the tracking error e exponentially decreases.*

Proof: By the intermediate value theorem, there exists $\beta \in [0, u]$ such that eqn. (2.9) can be written as

$$\dot{e} = \Lambda + h(x,0) + \frac{\partial h(x,\beta)}{\partial u} u. \tag{2.12}$$

For $|e| > \sigma$, select the Lyapunov function

$$V = \frac{1}{2} e^2. \tag{2.13}$$

Differentiating V along the solution of (2.12) with the control (2.11) yields

$$\begin{aligned} \dot{V} &= e[\Lambda + h(x,0)] - \frac{e}{\epsilon_0}\frac{\partial h(x,\beta)}{\partial u}\left[Ke + |\Lambda|\text{sign}(e) + b\ \text{sign}(e)\right] \\ &\leq -\frac{Ke^2}{\epsilon_0}\frac{\partial h(x,\beta)}{\partial u} - \left[\frac{1}{\epsilon_0}\frac{\partial h(x,\beta)}{\partial u} - 1\right]|e|(|\Lambda| + b) \\ &\leq -\frac{Ke^2}{\epsilon_0}\frac{\partial h(x,\beta)}{\partial u} \leq -Ke^2. \end{aligned} \tag{2.14}$$

Therefore, e exponentially decreases if $|e| > \sigma$. ■

Due to the exponential decrease proved in Lemma 2.2.2, from any initial condition $e(0)$, the condition $|e(t)| \leq \sigma$ is achieved in finite time. Once the condition $|e(t)| \leq \sigma$ is achieved, it will be maintained for all future times. In the following, we mainly discuss the case that $|e| \leq \sigma$.

If $|e| \leq \sigma$, substituting (2.11) into (2.9) yields

$$\dot{e} = -Ke + (\Delta(x,u) - u_{ad}) - \epsilon_f \text{sat}\left(\frac{e}{\mu_e}\right) \tag{2.15}$$

where

$$\Delta(x,u) = h(x,u) - cu.$$

If

$$\Delta(x,u) - u_{ad} = 0,$$

it is easy to prove that e converges to zero. Lemma 2.2.3 shows that under suitable assumptions there indeed exists a unique $u_{ad} = f(x)$ such that

$$\Delta(x,u) - f(x) = 0. \tag{2.16}$$

Lemma 2.2.3 *For eqn. (2.16) with u defined by the first equation in (2.11) and $u_{ad} = f(x)$, under Assumption 2.1.1, there exits a unique continuous $f(x)$ such that $f(x)$ satisfies (2.16) for all $x \in \mathcal{D}^n$.*

Proof: Noting Assumption 2.1.1, Lemma 2.2.3 can be proved by following the proof of Lemma 1 in [9, 49]. ■

It should be noted that in (2.16) u is defined by the first equation in (2.11) with $u_{ad} = f(x)$. The existence of $f(x)$ is guaranteed by the inequalities in (2.4). However, we cannot obtain the explicit expression of f because $h(x,u)$ is unknown. In the sequel, we apply a self-organized locally weighted learning algorithm (LWL) to develop a basis for and an approximation to $f(x)$. The importance of Lemma 2.2.3 is that f is not a explicit function of u. This allows (2.11) to be written as an explicit function of x because u_{ad} will only be a function of x not u.

2.3 LWL Algorithm and Structure Adaptation

In LWL [4, 21, 64, 76], the approximation to $f(x)$ at a point x is formed from the normalized weighted average of local approximators $\hat{f}_k(x)$ such that

$$\hat{f}(x) = \frac{\sum_k \omega_k(x)\hat{f}_k(x)}{\sum_k \omega_k(x)} \tag{2.17}$$

where each ω_k is nonzero only on a set denoted by S_k (defined below in eqn. (2.18)) over which $\hat{f}_k$ will be adapted to improve its accuracy relative to f.

2.3.1 Weighting Functions

We define a continuous, non-negative and locally supported weighting function $\omega_k(x)$ for the k-th local approximator. Denote the support of $\omega_k(x)$ by

$$S_k = \Big\{ x \in \mathcal{D}^n \mid \omega_k(x) \neq 0 \Big\}. \tag{2.18}$$

Let $\bar{S}_k$ denote the closure of S_k. Note that $\bar{S}_k$ is a compact set. An example of a weighting function satisfying the above conditions is the biquadratic kernel defined as

$$\omega_k(x) = \begin{cases} \left(1 - \left(\frac{||x-c_k||}{\mu}\right)^2\right)^2, & \text{if } ||x - c_k|| < \mu \\ 0, & \text{otherwise.} \end{cases} \tag{2.19}$$

where c_k is the center location of the k-th weighting function and μ is a constant which represents the radius of the region of support. In this example, the region of support is

$$S_k = \Big\{ x \in \mathcal{D}^n \mid \|x - c_k\| < \mu \Big\}. \tag{2.20}$$

Since the approximator is self-organizing, the number of local approximators $N(t)$ is not constant. Conditions for increasing N at discrete instants of time are presented in

Section 2.3.3. Since N is time varying, the region over which the approximator defined in eqn. (2.17) can have a nonzero value is also time varying. This region is defined as

$$\mathcal{A}^{N(t)} = \bigcup_{1 \le k \le N(t)} S_k.$$

When $x(t) \in \mathcal{A}^{N(t)}$, there exists at least one k such that $\omega_k(x) \neq 0$. The normalized weighting functions are defined as

$$\bar{\omega}_k(x) = \frac{\omega_k(x)}{\sum_{k=1}^{N(t)} \omega_k(x)}.$$

The set of non-negative functions $\{\bar{\omega}_k(x)\}_{k=1}^{N(t)}$ forms a *partition of unity* on $\mathcal{A}^{N(t)}$:

$$\sum_{k=1}^{N(t)} \bar{\omega}_k(x) = 1, \quad \text{for all } x \in \mathcal{A}^{N(t)}.$$

Note that the support of $\omega_k(x)$ is exactly the same as the support of $\bar{\omega}_k(x)$.

When $x(t) \notin \mathcal{A}^{N(t)}$, all $\omega_k(x)$ for $1 \le k \le N(t)$ are zero. Therefore, to complete the approximator definition of eqn. (2.17) to be valid for any $x \in \Re^n$:

$$\hat{f}(x) = \begin{cases} \sum_{k=1}^{N(t)} \bar{\omega}_k(x) \hat{f}_k(x) & \text{if } x \in \mathcal{A}^{N(t)} \\ 0 & \text{if } x \in \Re^n - \mathcal{A}^{N(t)}. \end{cases} \tag{2.21}$$

In the reminder of this section, we will only consider the case when $x(t) \in \mathcal{A}^{N(t)}$ to give all definitions for the LWL algorithm.

2.3.2 Local Approximators

We define

$$\hat{f}_k(x) = \Phi_k^T \hat{\theta}_{f_k} \tag{2.22}$$

where Φ_k is a prespecified vector of continuous basis functions. For the function $f(x)$, the vector $\theta^*_{f_k}$ denotes the unknown optimal parameter estimates for $x \in \bar{S}_k$:

$$\theta^*_{f_k} = \arg\min_{\hat{\theta}_{f_k}} \left(\int_{\bar{S}_k} \omega_k(x) \left| f(x) - \hat{f}_k(x) \right|^2 dx \right). \tag{2.23}$$

Note that $\theta^*_{f_k}$ is well defined for each k because f and $\hat{f}_k$ are smooth on compact $\bar{S}_k$. Therefore,

$$f^*_k = \Phi_k^\top \theta^*_{f_k}$$

will be referred to as the optimal local approximator to f on $\bar{S}_k$.

Let the optimal approximation error to f on $\bar{S}_k$ be denoted as ϵ_{f_k}:

$$\epsilon_{f_k}(x) \quad = \quad f(x) - f^*_k(x). \tag{2.24}$$

Since in subsequent expressions ϵ_{f_k} only appears as a product with $\omega_k(x)$, the value of $\epsilon_{f_k}(x)$ is immaterial outside $\bar{S}_k$. In order for ϵ_{f_k} to be defined everywhere, let

$$\epsilon_{f_k}(x) = \begin{cases} f(x) - f^*_k(x), & \text{on } \bar{S}_k, \\ 0, & \text{otherwise.} \end{cases}$$

The controller will use a known design constant $\epsilon_f > 0$. We make the following assumption.

Assumption 2.3.1 *The basis set Φ_k and μ are selected such that $|\epsilon_{f_k}(x)| \leq \bar{\epsilon}_f$ for $x \in \bar{S}_k$ for some (unknown) positive constant $\bar{\epsilon}_f < \epsilon_f$.*

For a linear basis set $\Phi_k^\top = [1, x - c_k]$ for $k \in [1, N]$ when the region of supports are chosen as (2.20), this assumption is satisfied if

$$|f''(x)| < \frac{\epsilon}{2\mu^2}$$

for $x \in S_k$. Note that the boundedness of $\max_{x\in\bar{S}_k}(|\epsilon_{f_k}(x)|)$ comes from the fact that $|\epsilon_{f_k}|$ is continuous on compact $\bar{S}_k$.

For any $x \in \mathcal{A}^{N(t)}$, $f(x)$ can be represented as the weighted sum of the local optimal approximators:

$$f(x) = \sum_k \bar{\omega}_k(x) f_k^*(x) + \delta_f(x). \tag{2.25}$$

This expression defines the optimal approximation error $\delta_f(x)$ on $\mathcal{A}^{N(t)}$ which satisfies $|\delta_f(x)| \leq \bar{\epsilon}_f$, since

$$\begin{aligned} |\delta_f| &= \left| f(x) - \sum_k \bar{\omega}_k(x) f_k^*(x) \right| = \left| \sum_k \bar{\omega}_k(x)(f(x) - f_k^*(x)) \right| \\ &\leq \sum_k \bar{\omega}_k(x) |\epsilon_{f_k}(x)| \qquad (2.26) \\ |\delta_f| &\leq \max_k(|\epsilon_{f_k}|) \sum_k \bar{\omega}_k(x) = \bar{\epsilon}_f. \qquad (2.27) \end{aligned}$$

Therefore, if each local optimal model $f_k(x)$ has accuracy $\bar{\epsilon}_f$ on $\bar{S}_k$, then the global accuracy of $\sum_k \bar{\omega}_k(x) f_k(x)$ on $\mathcal{A}^{N(t)}$ also achieves at least accuracy $\bar{\epsilon}_f$. The δ_f term in (2.25) is the *inherent approximation error* of $\hat{f}(x)$ for $f(x)$.

For the adaptive portion of the control law we choose

$$u_{ad} = \hat{f}. \tag{2.28}$$

To obtain $\hat{f}$, we need to estimate θ_{f_k}. For $x \in \mathcal{A}^i$, we choose the adaptive law

$$\dot{\hat{\theta}}_{f_k} = \begin{cases} \Gamma_{f_k} \bar{\omega}_k e \Phi_k & \text{if } |e| > \mu_e \\ 0 & \text{otherwise} \end{cases} \tag{2.29}$$

where $\mu_e(< \sigma)$ is a positive constant. The parameter adaptation will turn off when either $x \notin \mathcal{A}^i$ or $|e| \leq \mu_e$. This choice of adaption law is justified in eqn. (2.37).

2.3.3 Structure Adaptation

We initialize $\hat{f}$, the approximation to f in (2.21), with no local approximators, i.e., $N(0) = 0$; therefore, the set $\mathcal{A}^0$ is initially empty. We define the following criteria for adding a new local approximator to the approximation structure. A local approximator $\hat{f}_k$ is added and $N(t)$ is increased by one:

1. if $|e| \leq \sigma$ and the present operating point $x(t)$ does not activate any of the existing local approximators (i.e., $\max_{1\leq k\leq N(t)}(\omega_k(x)) = 0$); and
2. the function $\dot{V}_0(t) \geq 0$ (V_0 is defined in (2.34)) while $\mu_e < |e(t)| \leq \sigma$.

With the above criterions, $N(t)$ is non-decreasing. $\mathcal{A}^{N(t)}$ changes as $N(t)$ increases. Therefore, the structure of $\hat{f}$ in (2.21) changes as $N(t)$ increases. This criteria is motivated following eqn. (2.36).

2.4 Self-organizing Control and Stability Analysis

For the controller described in Sections 2.2-2.3, we have the following result.

Theorem 2.4.1 *The system described by eqn. (2.1–2.2) with control law*

$$u = \begin{cases} \dfrac{1}{c}\left(-Ke - \Lambda - \hat{f} - \epsilon_f \mathrm{sat}\left(\dfrac{e}{\mu_e}\right)\right), & |e| \leq \sigma \\ -\dfrac{1}{\epsilon_0}[Ke + (|\Lambda| + b(x))\mathrm{sign}(e)], & |e| > \sigma \end{cases} \tag{2.30}$$

using the self-organizing function approximation (2.21) with update laws (2.29) and structure adaptive criterion in Subsection 2.3.3 has the following properties:

1. $\tilde{x}$, e, $\tilde{\theta}_{f_k}$, $\hat{\theta}_{f_k}$, $N(t) \in \mathcal{L}_\infty$;

2. $e(t) = L^\top \tilde{x}$ *is ultimately bounded by* $|e(t)| \leq \mu_e$*;*

3. each $\tilde{x}_i$ *is ultimately bounded by* $|\tilde{x}_i| \leq 2^{i-1}\lambda^{i-n}\mu_e$*, for* $i = 1, \ldots, n$*, with* λ *being a constant selected for designing the* L *vector,*

where $\tilde{\theta}_{f_k} = \hat{\theta}_{f_k} - \theta^*_{f_k}$.

In the proof, we will use the following notation. For $i \geq 1$, we denote the time at which the i-th local model is added as T_i (i.e., $N(T_i) = i$ and $\lim_{\epsilon \to 0} N(T_i - \epsilon) = i - 1$). With this notation, $N(t)$ is constant with value i for $t \in [T_i, T_{i+1})$. The center location of the new local approximator c_i is defined such that $x(T_i) \in S_i$ and $c_j \notin S_i$ for $j \in [1, i)$. It is possible that for some i, the approximator has sufficient approximation capability, in which case $T_{i+1} = \infty$. In the proof, the analysis will be concerned with bounding the duration of time for which $|e(t)| > \mu_e > 0$. Therefore, to decrease redundancy later, we introduce the function $\bar{\mu}(e, \mu_e, t_1, t_2)$ which measures the amount of time in the interval $[t_1, t_2]$ for which $|e(t)| > \mu_e$. For example, this function could be computed as

$$\bar{\mu}(e, \mu_e, t_1, t_2) = \int_{t_1}^{t_2} 1\left(|e(t)| - \mu_e\right) dt, \tag{2.31}$$

where the function $1(\lambda)$ is defined as

$$1(\lambda) = \begin{cases} 1 & \text{if } \lambda > 0 \\ 0 & \text{otherwise.} \end{cases}$$

Proof: If $|e| > \sigma$, with control law (2.30) we have proven that e will exponentially decrease. In the following, we only consider the case that $|e| \leq \sigma$. With the control law defined in eqn. (2.11), for $x \in \mathcal{D}^n$, the closed-loop dynamic equation derived from (2.6), using (2.15), (2.16), and (2.28), is

$$\dot{e} \;\; = \;\; -Ke + (f - \hat{f}) - \epsilon_f \text{sat}\left(\frac{e}{\mu_e}\right). \tag{2.32}$$

The proof is completed in three steps.

A. Analysis for $x \notin \mathcal{A}^{N(t)}$

We will first consider the tracking performance for $x \notin \mathcal{A}^{N(t)}$ (i.e., $\hat{f} = 0$). In this case, the closed-loop dynamic equation for e defined in (2.32) becomes

$$\dot{e} = -Ke + f - \epsilon_f \text{sat}\left(\frac{e}{\mu_e}\right). \tag{2.33}$$

For $x(t) \notin \mathcal{A}^{N(t)}$, define the Lyapunov function as

$$V_0(t) = V_0(e(t)) = \frac{1}{2}e^2. \tag{2.34}$$

Then the derivative of V_0 along solutions of (2.33) is

$$\dot{V}_0 = -Ke^2 + e\Big(f - \epsilon_f \text{sat}\left(\frac{e}{\mu_e}\right)\Big). \tag{2.35}$$

When $|e(t)| > \mu_e$ and $|f(x)| \leq \epsilon_f$, then

$$e\Big(f - \epsilon_f \text{sat}\left(\frac{e}{\mu_e}\right)\Big) \leq 0.$$

The derivative of $V_0(t)$ is reduced to

$$\dot{V}_0 \;\; \leq \;\; -Ke^2 = -2KV_0 < 0. \tag{2.36}$$

Therefore, if $|f(x)| \leq \epsilon_f$ while $|e(t)| > \mu_e$, then the Lyapunov function V_0 must decrease. If V_0 increases while $|e(t)| > \mu_e$, then it must be true that $|f(x)| > \epsilon_f$. This motivates the structure adaptation criteria defined in Section 2.3.3.

B. Analysis for $t \in [T_i, T_{i+1})$

The goal of this section is to prove that for $t \in [T_i, T_{i+1})$ (i.e., the number of local regions over this time interval is fixed to i), then $\tilde{x}$, e, $\tilde{\theta}_{f_k}$, $\hat{\theta}_{f_k} \in \mathcal{L}_\infty$ for $k = 1, \ldots, i$ and that the total time is bounded for which $|e(t)| > \mu_e$. To simplify the notation, we use the fact that $i = N(T_i)$ and define $T_{i+1}^- = \lim_{\epsilon \to 0}(T_{i+1} - \epsilon)$.

As shown in Section 2.3.2, the approximation

$$f^*(x) = \sum_k \bar{\omega}_k(x) f_k^*(x)$$

achieves at least accuracy ϵ_f on $\mathcal{A}^i$ (i.e., $|f(x) - f^*(x)| \le \epsilon_f$ for any $x \in \mathcal{A}^i$). The following analysis considers the adaptation of $\hat{f}$ to achieve the tracking specification for $x \in \mathcal{A}^i$.

For $x(t) \in \mathcal{A}^i$, we consider the Lyapunov function

$$V_i(t) \;=\; \frac{1}{2}e^2 + \frac{1}{2}\sum_{k=1}^{i} \tilde{\theta}_{f_k}^\top \Gamma_{f_k}^{-1} \tilde{\theta}_{f_k} = V_0 + V_\theta^i$$

where

$$V_\theta^i = \frac{1}{2}\sum_{k=1}^{i} \tilde{\theta}_{f_k}^\top \Gamma_{f_k}^{-1} \tilde{\theta}_{f_k}$$

and the positive definite matrices $\Gamma_{f_k}, k = 1, \cdots, i$ represent learning rates.

Let $t \in [t_1, t_2] \subset [T_i, T_{i+1})$ denote a time interval for which $e(t)$ is outside the deadzone (i.e., $|e(t)| \ge \mu_e$). Over this time interval, the state $x(t)$ could be either outside $\mathcal{A}^i$ or inside $\mathcal{A}^i$.

1. For any subinterval $t \in [\tau_1, \tau_2] \subset [t_1, t_2]$ for which $x(t) \notin \mathcal{A}^i$, the parameter adaptation will automatically stop because $\bar{\omega}_k(x) = 0 \ \forall k = 1, \ldots, i$. $V_\theta^i(t)$ is constant over this time interval. Therefore, according to (2.36),

$$\dot{V}_i(t) = \dot{V}_0(t) \le -Ke^2$$

and V_i decreases for $t \in [\tau_1, \tau_2]$, such that $V_i(\tau_2) \leq V_i(\tau_1)$.

2. For any subinterval $t \in [\tau_2, \tau_3] \subset [t_1, t_2]$ for which $x(t) \in \mathcal{A}^i$, the time derivative of $V_i(t)$ along the solution of (2.32) is:

$$
\begin{aligned}
\dot{V}_i(t) &= e\Big(f - \hat{f} - \epsilon_f \mathrm{sat}\left(\frac{e}{\mu_e}\right)\Big) + \sum_{k=1}^{i} \tilde{\theta}_{f_k}^\top \Gamma_{f_k}^{-1} \dot{\theta}_{f_k} - Ke^2 \\
&= e\sum_{k=1}^{i} \overline{\omega}_k \Phi_k^\top (\theta_{f_k}^* - \theta_{f_k}) + e\delta_f - e\epsilon_f \mathrm{sat}\left(\frac{e}{\mu_e}\right) + \sum_{k=1}^{i} \left(\tilde{\theta}_{f_k}^\top \Gamma_{f_k}^{-1} \dot{\theta}_{f_k}\right) - Ke^2 \\
&= -Ke^2 + e\delta_f - e\epsilon_f \mathrm{sat}\left(\frac{e}{\mu_e}\right) + \sum_{k=1}^{i} \tilde{\theta}_{f_k}^\top \Gamma_{f_k}^{-1} \Big(\dot{\hat{\theta}}_{f_k} - \Gamma_{f_k} e \overline{\omega}_k \Phi_k\Big). \qquad (2.37)
\end{aligned}
$$

Substituting (2.29) into (2.37), we obtain

$$
\dot{V}_i(t) \leq -Ke^2 \qquad (2.38)
$$

for any $t \in [\tau_2, \tau_3]$.

Therefore, $\forall t \in [t_1, t_2]$ with $|e(t)| > \mu_e$, we have shown that

$$
\dot{V}_i(t) \leq -Ke^2 < -K\mu_e^2 < 0. \qquad (2.39)
$$

From this, it is straightforward to show that for $\bar{\mu}$ defined in eqn. (2.31)

$$
V_i(t_2) - V_i(t_1) \leq -K\mu_e^2(t_2 - t_1) = -K\mu_e^2 \bar{\mu}(e, \mu_e, t_1, t_2)
$$

which implies

$$
\bar{\mu}(e, \mu_e, t_1, t_2) \leq \frac{V_i(t_1) - V_i(t_2)}{K\mu_e^2} \leq \frac{V_i(t_1)}{K\mu_e^2}.
$$

Next, we consider the case where $e(t)$ enters the deadzone at time t_2, stay inside the deadzone until $t_3 \geq t_2$, and leaves the deadzone at t_3. Therefore, $t \in [t_2, t_3] \subset$

$[T_i, T_{i+1})$ denotes a time interval for which $|e(t)| \leq \mu_e$ and $N(t)$ is constant. Therefore, $\bar{\mu}(e, \mu_e, t_2, t_3) = 0$. In addition, the following facts are true: (1) on the interval $[t_2, t_3]$, the approximator parameters are constant (i.e., adaptation is off); (2) $|e(t_2)| = |e(t_3)| = \mu_e$; and, (3) $|e(t)| \leq |e(t_3)|,\ \forall t \in [t_2, t_3]$. Using these facts, it is straightforward to show that $V_i(t_2) = V_i(t_3)$ and $V_i(t) \leq V_i(t_3),\ \forall t \in [t_2, t_3]$. Note that these facts are independent of whether $x(t)$ enters or leaves the region $\mathcal{A}^i(t)$ over the time interval $[t_2, t_3]$.

This paragraph will consider the stability properties for any $t \in [T_i, T_{i+1})$. According to the criteria given in Section 2.3.3 for adding the ith local region, $|e(t)| > \mu_e$ at $t = T_i$. Let $t_1 = T_i$. Note that $e(t_1)$ is outside the deadzone. Assume that $e(t)$ enters the deadzone at t_{2j}, leaves at t_{2j+1}, for $j \geq 1$, and eventually stays outside the deadzone until T_{i+1}^-. Let $\bar{t} \in [T_i, T_{i+1})$ be the last time in this interval such that $|e(\bar{t})| \leq \mu_e$. Therefore, the total time outside the deadzone for $t \in [T_i, T_{i+1})$ is

$$\begin{aligned}
\bar{\mu}(e, \mu_e, T_i, T_{i+1}^-) &= \sum_{j \geq 1} (t_{2j} - t_{2j-1}) + (T_{i+1}^- - \bar{t}) \\
&\leq \frac{1}{K\mu_e^2} \left[\sum_{j \geq 1} \Big(V_i(t_{2j-1}) - V_i(t_{2j}) \Big) + \Big(V_i(\bar{t}) - V_i(T_{i+1}^-) \Big) \right] \\
&= \frac{1}{K\mu_e^2} \Big(V_i(T_i) - V_i(T_{i+1}^-) \Big). \qquad (2.40)
\end{aligned}$$

Eqn. (2.40) together with the facts that $V_i(T_i)$ is finite and $V_i(T_{i+1}^-) < V_i(T_i)$ proves that on each interval $[T_i, T_{i+1})$, the total time outside the deadzone is finite. Therefore, either T_{i+1} is infinite with $|e(t)|$ ultimately bounded by μ_e, or T_{i+1} is finite with $N(t)$ increased by one at $t = T_{i+1}$.

The fact that $\forall\, t \in [T_i, T_{i+1})$, $V_i(t) \leq V_i(T_i)$ follows directly from previous analysis whether $|e(t)| > \mu_e$ or $|e(t)| \leq \mu_e$. Therefore, $\tilde{x}$, e, $\tilde{\theta}_{f_k}$, $\hat{\theta}_{f_k} \in \mathcal{L}_\infty(T_i, T_{i+1}^-)$. Note that these properties hold even if the state enters or leaves the region $\mathcal{A}^i$ an infinite number of times, or if the combined tracking error e enters or leaves the deadzone an infinite number of times.

Since $\mathcal{D}^n$ is compact, x is bounded. Assume N tends to infinity, then there exists a bounded sequence of center locations $\{c_i\}_{i=1}^{\infty}$ with each $c_i \in \mathcal{D}^n$. Any bounded infinite sequence on a compact set has a convergent subsequence. Let $\{c_{i_k}\}_{k=1}^{\infty}$ be a convergent subsequence of $\{c_i\}_{i=1}^{\infty}$, then there exists an integer L such that for $i_k > L$, $\|c_{i_k} - c_{i_{k-1}}\| < \mu$. But $\|c_i - c_j\| > \mu$ $(\forall i, j)$ by the structure adaptation. There is a contradiction. Therefore, N must be finite.

C. Analysis for $t \in [0, \infty)$

Let the time interval of operation be specified as $[T_0, T_f]$, where T_f can be infinite. We initialize the approximator structure with $N(T_0) = 0$. Denote the times at which $N(t)$ increases as T_i as discussed above.

When $t \in [T_0, T_1)$, $N(t) = 0$ and $\hat{f} = 0$. Either the total time such that $|e(t)| > \mu_e$ is less than $\frac{1}{K} ln \left(\frac{|e(T_0)|}{\mu_e} \right)$, $T_1 = \infty$, and the theorem is proved; or, T_1 is finite. In either case,

$$V_0(T_1^-) \leq \max \left(V_0(T_0), \frac{1}{2}\mu_e^2 \right).$$

For $i \geq 1$ the i-th local region is added at $t = T_i$. Next, we prove the boundedness of each $V_i(T_i)$ during the transition from $V_{i-1}(T_i^-)$ to $V_i(T_i)$, i.e., we want to show that each $V_i(T_i)$ has a finite value. The Lyapunov function at $t = T_i$ is

$$V_i(T_i) = \frac{1}{2}e^2(T_i) + \frac{1}{2}\sum_{k=1}^{i} \tilde{\theta}_{f_k}^{\top}(T_i)\Gamma_{f_k}^{-1}\tilde{\theta}_{f_k}(T_i)$$

Note that $e(T_i) = e(T_i^-)$, because $e(t)$ is continuous during the transition from T_i^- to T_i. Since $x(T_i)$ does not activate the first $(i-1)$ local approximators when the ith local approximator is added at $t = T_i$, the parameter estimates $\hat{\theta}_{f_k}$, $k = 1, \cdots, i-1$ are unchanged from $t = T_i^-$ to $t = T_i$. Thus,

$$
\begin{aligned}
V_i(T_i) &= \frac{1}{2}e^2(T_i^-) + \frac{1}{2}\sum_{k=1}^{i-1}\tilde{\theta}_{f_k}^\top(T_i^-)\Gamma_{f_k}^{-1}\tilde{\theta}_{f_k}(T_i^-) + \frac{1}{2}\tilde{\theta}_{f_i}^\top(T_i)\Gamma_{f_i}^{-1}\tilde{\theta}_{f_i}(T_i) \\
&= V_{i-1}(T_i^-) + \frac{1}{2}\tilde{\theta}_{f_i}^\top(T_i)\Gamma_{f_i}^{-1}\tilde{\theta}_{f_i}(T_i)
\end{aligned}
$$

For any $t \in [T_{i-1}, T_i^-]$, $V_{i-1}(t) \leq V_{i-1}(T_{i-1})$. Then we proceed to attain

$$
\begin{aligned}
V_i(T_i) &\leq V_{i-1}(T_{i-1}) + \frac{1}{2}\tilde{\theta}_{f_i}^\top(T_i)\Gamma_{f_i}^{-1}\tilde{\theta}_{f_i}(T_i) \\
&\leq \frac{1}{2}e(T_1)^2 + \frac{1}{2}\sum_{k=1}^{i}\tilde{\theta}_{f_k}^\top(T_k)\Gamma_{f_k}^{-1}\tilde{\theta}_{f_k}(T_k). \qquad (2.41)
\end{aligned}
$$

For $k = 1, \cdots, i$, each $\tilde{\theta}_{f_k}(T_k) = \hat{\theta}_{f_k}(T_k) - \theta_{f_k}^*$ has a finite value as long as the initial parameter estimate $\hat{\theta}_{f_k}(T_k)$ at $t = T_k$ is finite. Since only a finite number of increments of N can occur (i.e., $N(T_i) = i < \infty$), the summation term on the right of the inequality (2.41) has a finite value. The $e(T_1)$ term is also a finite value. This directly yields that $V_i(T_i) < \infty$, which implies that $\tilde{x}$, e, $\tilde{\theta}_{f_k}$, $\hat{\theta}_{f_k} \in \mathcal{L}_\infty$.

The ultimate bound on each $\tilde{x}_i$ comes directly from (2.7), which is guaranteed by the selection of L as discussed in [67]. ■

Remark 2.4.2 *In Theorem 2.4.1, the proposed control law consists of two parts. When* $|e| > \sigma$*, the control law is a sliding mode control. The magnitude of the control depends on the bound function* $b(x)$*. When* $|e| \leq \sigma$*, the control law is a self-organizing control. The self-organizing control learns the unknown system and improves the tracking performance* *gradually based on the structure adaptation scheme. In the self-organizing control the magnitude of the self-organizing control does not depend on the bound function $b(x)$. Fig. 2.1 shows as a solid line a sketch of the magnitude of the switching term in the control law in Theorem 2.4.1. At the switching hyperplane $|e| = \sigma$, the controller (2.30) is discontinuous. During the control, at some finite time the controller may be switching. The proof of Theorem 1 shows that $|e| \leq \mu_e$ as time tends to infinity. Therefore, as time tends to infinity the controller is not switching. In the control, several control parameters play important roles. Constant σ determines the size of the operational region $\mathcal{D}^n$ to which the self-organizing control applies. If σ is large, the size of the region $\mathcal{D}^n$ will be large. Constant $\mu_e (< \sigma)$ determines the ultimate bound on the tracking error. During the control, μ determines the size of the locally learning region. If μ is small, ϵ_f can be chosen small. But small μ means that there will potentially be a large number of local learning regions. In practical control, the control parameters should be chosen according to the tradeoff of different factors.*

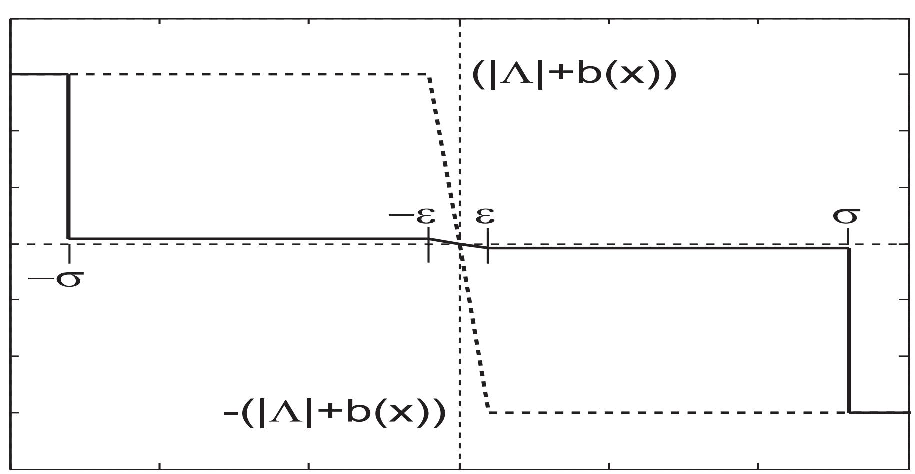

Figure 2.1: Sketch of the magnitude of the switching term in the control laws of eqns. (2.43) (dotted) and (2.30) (solid) vs e.

Remark 2.4.3 *The tracking control problem could also be solved by*

$$u = -\frac{1}{\epsilon_0}\left(Ke + (|\Lambda| + b(x))\text{sign}(e)\right) \tag{2.42}$$

with e *and* Λ *defined in (2.6) and (2.10), respectively; or by*

$$u = -\frac{1}{\epsilon_0}\left(Ke + (|\Lambda| + b(x))\text{sat}(e/\varepsilon)\right) \tag{2.43}$$

where ε *is a positive constant. The control of (2.42) achieves finite time convergence of* e *to zero and asymptotic convergence of* $x(t)$ *to* $x_c(t)$*, but requires control with magnitude* $|\Lambda|+b(x)$ *switching at very high rates. The control of (2.43) achieves finite time convergence to* $|e| \leq \varepsilon$ *and* $\|x - x_c\| < \gamma\varepsilon$ *where* γ *is a constant determined by the choice of* L *in (2.6). The control of (2.43) still requires control with magnitude* $|\Lambda| + b(x)$ *and within the region of the sat function the effective gain is* $(\|\Lambda\|+b)/\varepsilon$ *which can be quite large when the desired accuracy* ε *is small. A magnitude of the switching term in the* u *from (2.43) as a function of* e *is shown as the dotted line in Fig. 2.1. In this paper we have proposed a new approach. We introduce a new parameter* $\sigma > 0$ *which is finite, but can be significantly larger than* μ_e*. A large magnitude sliding mode term will be designed (see eqn. (2.11)) to ensure that* $|e(t)| \leq \sigma$ *is achieved in finite time and maintained for all future times. When* $|e(t)| \leq \sigma$ *a self-organizing approximation based controller is designed to ultimately achieve* $|e| \leq \mu_e$*. Due to the inclusion of the self-organizing approximator, the control is able to achieve this same tracking accuracy using a term of magnitude* ϵ_f *which is small in comparison to the switching term in control law (2.42). Fig. 2.1 shows as a solid line a sketch of the magnitude of the switching term from the proposed approach of (2.11).*

2.5 Simulation

We consider the following example for illustrative purpose.

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= h(x,u)\end{aligned}$$

where $x = [x_1, x_2]^\top$, $h(x,u)$ is unknown during the control. For simulation, we assume that

$$h(x,u) = \sin(0.4(x_1 + x_2)) + 2u + \sin(u).$$

Obviously, $h(x,u)$ satisfies Assumption 2.1.1 with $\epsilon_0 = 1$, $c(x) = 2$, and $b(x) = 2$.

Given a desired trajectory $[x_d(t), x_d^{(1)}(t), x_d^{(2)}(t)] = [z_1, z_2, z_3]$ which is generated by the third order system

$$\begin{aligned}\dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ \dot{z}_3 &= a_1(r - z_1) - a_2 z_2 - a_3 z_3\end{aligned}$$

where $[a_1, a_2, a_3] = [27, 27, 9]$ such that the transfer functions

$$\frac{x_d^{(i)}}{r} = \frac{27 s^i}{s^3 + 9s^2 + 27s + 27}, \; i = 0, 1, 2$$

are BIBO stable. As long as $r(t)$ is bounded, signals $x_d, x_d^{(1)}, x_d^{(2)}$ will be continuous and bounded. The main idea of such a prefilter approach to generating the reference trajectory is that the user specifies an arbitrary signal r. The prefilter computes the necessary derivatives and ensures that Assumption 2.1.1 is satisfied. Theoretically, r can be any bounded signal. For the purpose of this simulation r is selected to be $3\sin(0.1\pi t)$.

The tracking accuracy we specify to be achieved is that as $t \to \infty$, $|\tilde{x}_1| \leq \mu_x = 0.02$ with control gain $K = 1$. The linearly combined tracking error is defined as $e = L^\top \tilde{x} = [\lambda, 1]\tilde{x}$ with λ selected to be 1. The desired tracking accuracy for $e(t)$ is selected to be $|e| \leq \mu_e = 0.02$ so that we can ensure that $|\tilde{x}_1| \leq \frac{1}{\lambda}\mu_e \doteq \mu_x = 0.02$ ultimately. The sliding mode control is in charge if $|e| > \sigma = 1$. The function approximation accuracies are specified as $\epsilon_f = 0.03$.

The weighting function is the biquadratic kernel of the form

$$\omega_k(x) = \begin{cases} \left(1 - R^2\right)^2, & \text{if } R < 1 \\ 0, & \text{otherwise.} \end{cases} \tag{2.44}$$

where

$$R = \left\| \frac{|x_1 - c_{k,1}|}{\mu}, \frac{|x_2 - c_{k,2}|}{\mu} \right\|_\infty .$$

For this example, as done by the authors of [4, 21, 45, 76, 77], we specify the local basis function as $\bar{x}_k = [1, x_1 - c_{k,1}, x_2 - c_{k,2}]^\top$ with c_k being the center of the $\bar{S}_k$. Therefore, f_k^* is the optimal local affine approximation to f on $\bar{S}_k$. We select $\mu = 0.4$. The parameter estimate on the first allocated region, $\hat{\theta}_{f_1}$, will be initialized at $t = T_1$ as $\hat{\theta}_{f_1}(T_1) = [0, 0, 0]^\top$. When the k-th ($k \geq 2$) center is allocated at $t = T_k$, the initial parameter estimate $\hat{\theta}_{f_k}(T_k)$ is selected either to be zero vector or based on $\hat{\theta}_{f_j}(T_k)$, where j ($j < k$) is the index of the closest existing center to the k-th center. The logic for the parameter initialization is as follows:

if $(|c_{k,1} - c_{j,1}| < 1.5\mu)$ and $(|c_{k,2} - c_{j,2}| < 1.5\mu)$,

$$\begin{aligned}
\hat{\theta}_{f_k,1}(T_k) &= \bar{x}_j^\top \hat{\theta}_{f_j}(T_k) \\
\hat{\theta}_{f_k,2}(T_k) &= \hat{\theta}_{f_j,2}(T_k) \\
\hat{\theta}_{f_k,3}(T_k) &= \hat{\theta}_{f_j,3}(T_k)
\end{aligned}$$

else,

$$\hat{\theta}_{f_k}(T_k) = [0, 0, 0]^\top.$$

This initialization forces the k-th approximator to have the same value as the j-th approximator would have at the center of S_k. This is a means of "boot strapping" the learning process. The adaptation rate matrix is set to $\Gamma_{f_k} = \text{diag}([1, 1, 1])$ where $\text{diag}(v)$ is the square diagonal matrix with diagonal component equal to the vector v.

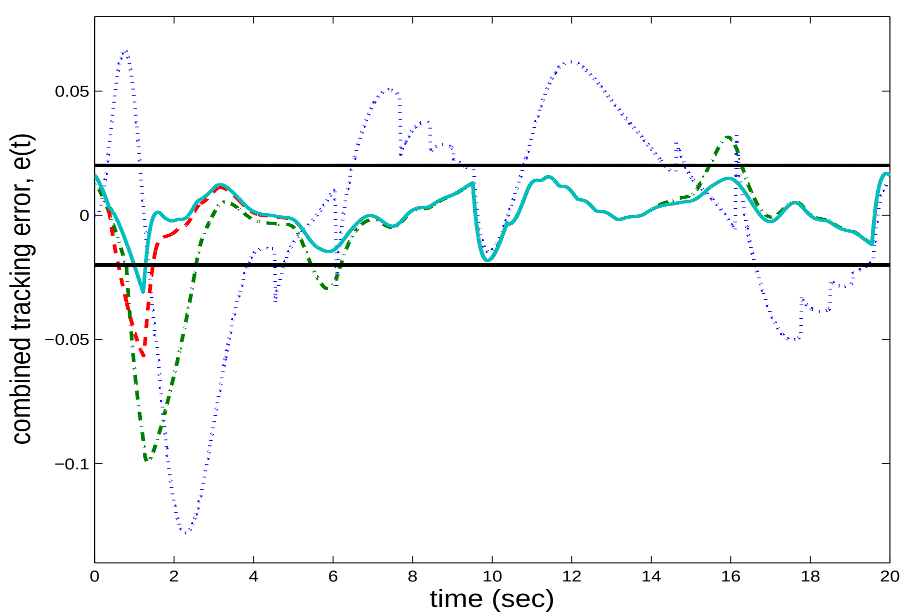

Figure 2.2: The combined tracking error $e(t)$ in the time interval of $[0, 20]$ (dotted), $[20, 40]$ (dash-dot), $[40, 60]$ (dashed) and $[60, 80]$ (solid).

Fig. 2.2 shows the performance of $e = \tilde{x}_1 + \tilde{x}_2$ for $t \in [0, 80]$. Since the period of the reference input is $T = 20$s, we have plotted $e(t)$ for $t \in [0, 20]$ (dotted), $t \in [20, 40]$ (dash-dot), $t \in [40, 60]$ (dashed) and $t \in [60, 80]$ (solid) along the same time axis. The time axis of each plot has been shifted by a multiple of the period $T = 20$ to increase the resolution of the time axis and to facilitate the comparison over repetitions of the trajectory. Note that with online approximation, since the local regions are being revisited periodically with $T = 20$, the tracking performance improves (i.e., $e(t)$ tends to decrease) with each repetition of the trajectory. This indicates the local approximators are learning to achieve increasingly more accurate approximation to the actual function. It is particularly important to note that the learning is a function of state. Therefore, the performance improvement will extend to different trajectories to the extent that the new trajectories utilize the same regions in state space. The improvement in tracking performance s shown in a different manner in Fig. 2.3, which plots the tracking errors $\tilde{x}_1$ and $\tilde{x}_2$ over the first four repetitions of the reference trajectory.

The number of local regions is depicted in Fig. 2.4. It can be seen that N increases with time and reaches its maximum at time $t \approx 19s$. After that N does not increase. Therefore, after $t = 19s$ no new local region is added.

The allocated center locations for $t \in [0, 21]$ are indicated on a phase plane plot of x_1 versus x_2 in Fig. 2.5. During the first 19 seconds, 17 centers are allocated. For the remaining simulation time, no more centers are added. Each '$\times$' indicates an allocated center location. The small square around each center indicates the support S_k of the corresponding local approximator. Fig. 2.5 also depicts the desired (dotted) and actual

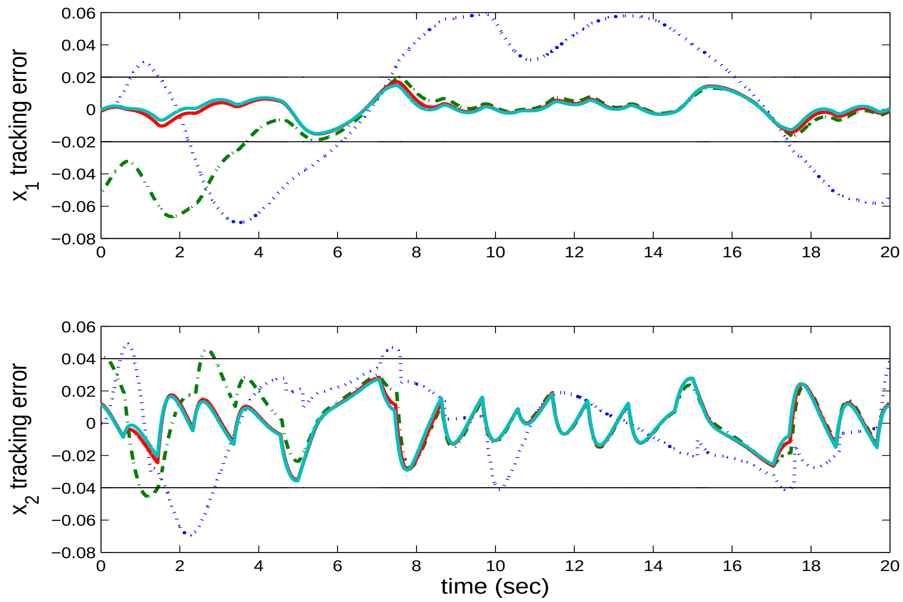

Figure 2.3: Plots of the tracking errors versus time t: (a) $\tilde{x}_1$ versus t; (b) $\tilde{x}_2$ versus t. Note that the horizontal axes are identical and that the caption is only applied to the (b) graph. The dotted lines are tracking errors in the time interval of $[0, 20]$.The dash-dot lines are tracking errors in the time interval of $[20, 40]$. The dashed lines are tracking errors in the time interval of $[40, 60]$. The solid lines are tracking errors in the time interval of $[60, 80]$.

(solid) trajectories over the time interval $[0, 80]$.

Fig. 2.6(a) plots the time outside the deadzone $\bar{\mu}(e, 0.02, t-20, t)$ at 20 second intervals, which is also the period of the commanded state $x_d(t)$. For example, consider $t = 20$, Fig. 2.6(a) shows that the time outside the deadzone in the interval $[t-20, t] = [0, 20]$ is approximately 16s. Prior to $t = 400$s, the combined tracking error e enters the deadzone and remains therein for the remainder of the simulation. Therefore, as shown in Fig. 2.6(b) the total time outside the deadzone is finite as t goes to infinity. This demonstrates that the tracking error is ultimately achieved using a self-organizing controller on $\mathcal{D}^n$ that does not include large gains or large-amplitude high-frequency switching.

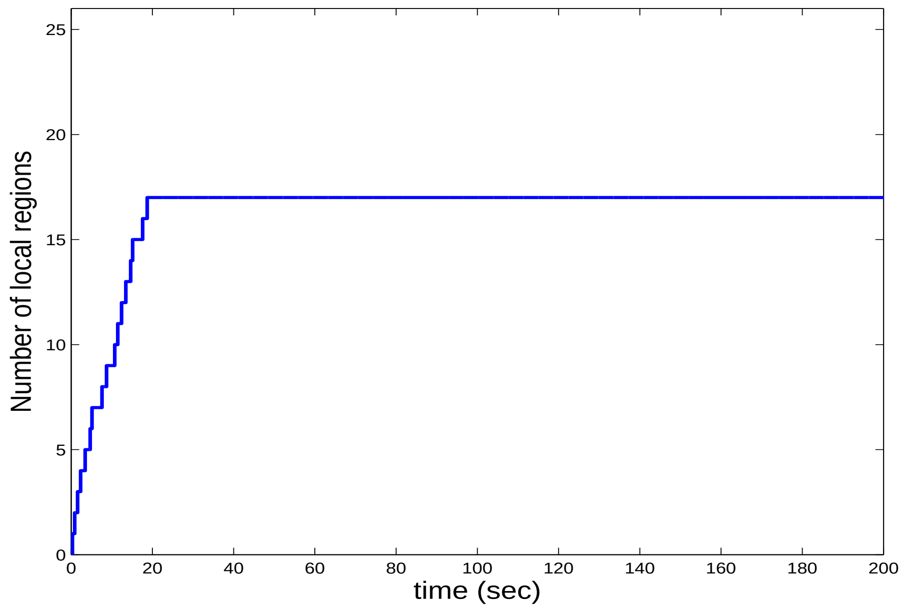

Figure 2.4: Number of local regions along time

2.6 Conclusion

This chapter considers tracking control for nonaffine systems (2.1)-(2.3). In the operational region a self-organizing controller is proposed with the aid of a lemma from [9, 49] and the self-organizing approach proposed in [77]. The proposed controller has the ability to adjust the structure of approximators and will make the tracking error smaller than a given positive constant. The approach can be directly extended to case where $\dot{x}_n = f(x) + g(x)u + h(x, u)$ where $f(x)$ and $g(x)$ are known, and $h(x, u)$ represents model

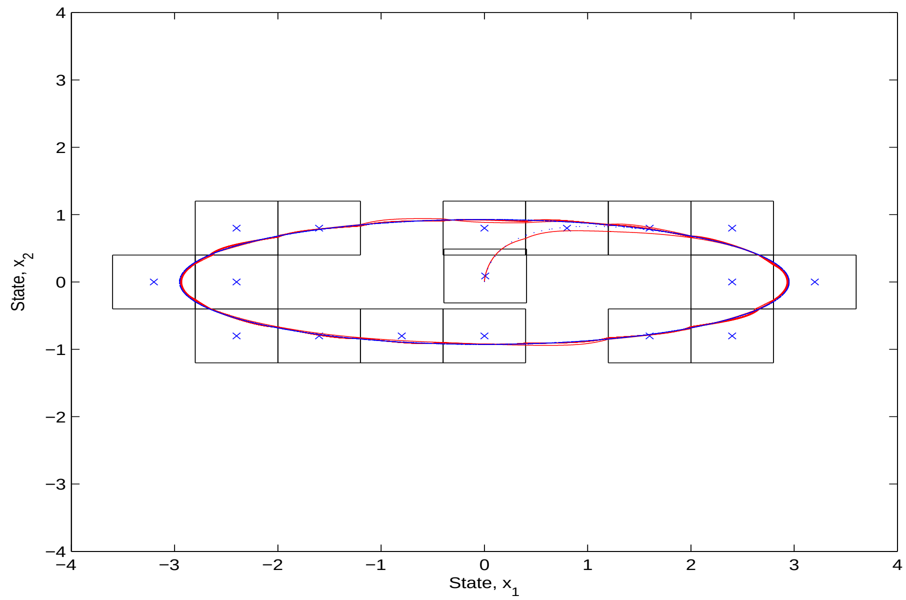

Figure 2.5: Phase plane plot of x_1 versus x_2 for $t \in [0, 80]$s. The dotted line is the desired trajectory for $t \in [0, 80]$s. The solid line shows the actual trajectory. The $\times$'s indicate the assigned center locations and the small square around each center location represents the associated region of support.

error. The approach can also be extended to case where

$$\begin{aligned}
\dot{x}_1 &= x_2 + \rho_1(x_1) \\
\dot{x}_3 &= x_3 + \rho_2(x_1, x_2) \\
&\vdots \\
\dot{x}_{n-1} &= x_n + \rho_{n-1}(x_1, \ldots, x_{n-1}) \\
\dot{x}_n &= f(x) + g(x)u + h(x, u)
\end{aligned}$$

where ρ_i and $h(x, u)$ are unknown, $f(x)$ and $g(x)$ are known. The simulation assumed a periodic input to simplify the presentation, but Theorem 2.4.1 makes no assumption regarding the periodicity of $x_c(t)$.

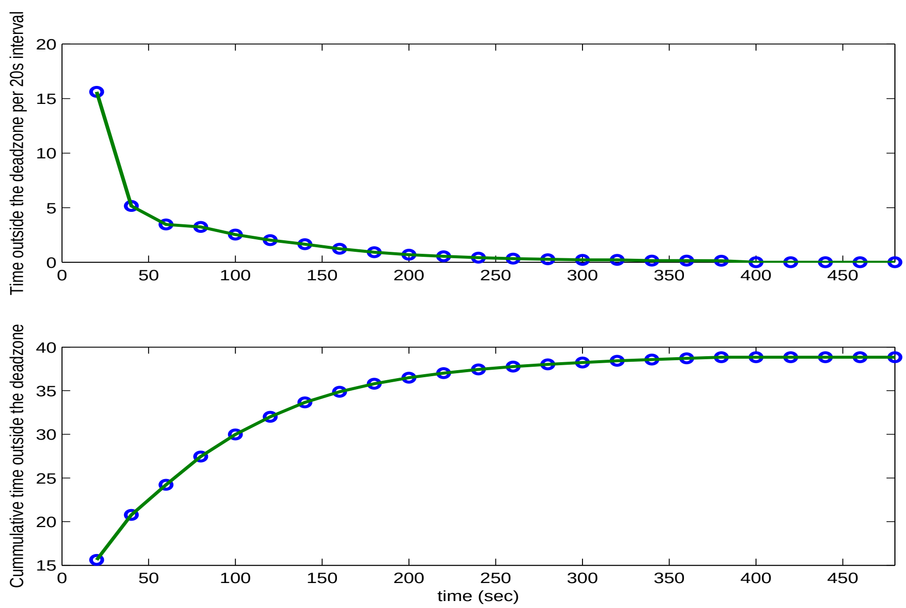

Figure 2.6: Graphs indicating the incremental and cumulative time that the tracking error e is outside the deadzone. (a) Each circle indicates the total time during the previous 20 second interval that the combined tracking error e was outside the deadzone (i.e., $|e| > 0.02$). (b) Cumulative time outside the deadzone. Note that the horizontal axes are identical and that the caption is only applied to the (b) graph.

Chapter 3

Point-wise Min-norm Optimal Control of Unknown Nonlinear Systems Based on Locally Weighted Learning

In this chapter, we consider optimal regulation and optimal tracking control of uncertain nonlinear systems. To deal with the uncertain terms, we propose locally weighted learning observers (LWLOs) to estimate the unknown nonlinear systems. Based on the approximators that result from locally weighted learning observers, point-wise min-norm optimal problems are defined for optimal regulation and optimal tracking control, respectively. Based on the defined optimal control problems, analytic controllers are proposed with the aid of control Lyapunov functions. Simulation results show effectiveness of the proposed control laws.

The organization of this chapter is as follows. In Sections 3.1 and 3.2, we discuss optimal regulation and tracking control problems and propose optimal controllers based on local weighted learning observers. In both sections, numerical examples are presented to show effectiveness of the proposed controllers.

3.1 Point-wise Min-norm Optimal Regulation

In this section, we consider the optimal regulation problem of the uncertain nonlinear system shown in (3.1).

3.1.1 Problem Statement

Consider an n-th order nonlinear system

$$\begin{aligned} \dot{x}_i &= x_{i+1}, \quad 1 \leq i \leq n-1 \\ \dot{x}_n &= f_0(x) + f(x) + g_0(x)u \end{aligned} \tag{3.1}$$

where $x = [x_1 \ldots, x_n]^\top \in R^n$ is the state, and $u \in R$ is the input. Functions f_0 and g_0 are known continuous functions. Function $f(x)$ is continuous in x and unknown. For a given bounded compact operational region $\mathcal{D}^n$ which includes the origin, $f(x)$ satisfies the following assumption.

Assumption 3.1.1 *The unknown function $f(x)$ satisfies $|f(x)| \leq h(x)$ for $x \in R^n - \mathcal{D}^n$, where $h(x)$ is a known function.*

Furthermore, function $g_0(x)$ satisfies the following standard controllability assumption.

Assumption 3.1.2 *Function $g_0(x)$ is bounded below, i.e.,*

$$g_0(x) > g_l(x) > c_g > 0$$

where c_g is a positive constant.

The problem discussed in this section is to design an optimal controller u such that the cost function

$$J_\infty = \int_0^\infty [q(x) + u^2] d\tau \tag{3.2}$$

achieves its minimum, where $q(x)$ is continuously differentiable and positive semi-definite.

If $f(x)$ is known, a standard dynamic programming argument reduces the above optimal control problem to finding the value function V^* solving the Hamilton-Jacobi-Bellman partial differential equation (HJB)

$$V_x^* f_e - \frac{1}{4}\left(V_x^* g_e g_e^\top V_x^{*\top}\right) + q(x) = 0 \tag{3.3}$$

where $f_e = [x_2, x_3, \ldots, x_n, f_0(x) + f(x)]^\top$, and $g_e = [0, \ldots, 0, g_0(x)]^\top$. If there exists a continuously differentiable positive definite solution $V^*(x)$ to eqn. (3.3), then the optimal controller is

$$u = -\frac{1}{2} g_e^\top V_x^{*\top}. \tag{3.4}$$

There are two obstacles which prevent us from finding the optimal controller. The first one is that $f(x)$ is unknown. The second one is that it is extremely difficult to solve the HJB partial differential equation (3.3) even if $f(x)$ is known. To overcome the first obstacle, we propose a locally weighted learning observer (LWLO) to estimate $f(x)$. To deal with the second obstacle, we modify the optimal problem to a new one such that analytic controllers can be proposed.

3.1.2 Locally Weighted Learning Observer

Let the observer be defined as follows.

$$\begin{aligned} \dot{\hat{x}}_i &= \hat{x}_{i+1}, \quad 1 \le i \le n-1 \\ \dot{\hat{x}}_n &= f_0(x) + \hat{f}(x) + g_0(x)u + v \end{aligned} \tag{3.5}$$

where $\hat{x} = [\hat{x}_1, \ldots, \hat{x}_n]^\top$ is the estimate of x, v is a stabilizing observer signal, $\hat{f}(x)$ is the estimate of $f(x)$ based on a locally weighted learning (LWL) algorithm [44, 45]. In LWL, the approximation to $f(x)$ at a point x are formed from the normalized weighted average of local approximators $\hat{f}_k(x)$ as last chapter such that

$$\hat{f}(x) = \frac{\sum_k \omega_k(x) \hat{f}_k(x)}{\sum_k \omega_k(x)} \tag{3.6}$$

where each ω_k is nonzero only on a set denoted by S_k (defined in Chapter 2) over which $\hat{f}_k$ will be adapted to improve its accuracy relative to f.

Let $z = [z_1, \ldots, z_n]^\top = x - \hat{x}$, we have

$$\begin{aligned} \dot{z}_i &= z_{i+1}, \quad 1 \le i \le n-1 \\ \dot{z}_n &= f(x) - \hat{f}(x) - v. \end{aligned} \tag{3.7}$$

Let

$$e(t) = L^\top z(t) \tag{3.8}$$

where

$$L = [l_1, l_2, \ldots, l_{n-1}, l_n]^\top = [\lambda^{n-1}, C_{n-1}^1 \lambda^{n-2}, \ldots, C_{n-1}^{n-2} \lambda, 1]^\top \tag{3.9}$$

λ is a positive constant, and $C_n^m = \frac{n!}{m!(n-m)!}$ for $1 \leq m \leq n-2$. We have the following lemma.

Lemma 3.1.3 *([77]) If* $\lim_{t\to\infty} |e(t)| \leq \mu_e$*, then* $\lim_{t\to\infty} |z_i| \leq \frac{2^{i-1}}{\lambda^{n-i}} \mu_e$ *for* $1 \leq i \leq n$*, where* μ_e *is a positive constant. Furthermore, if* $\lim_{t\to\infty} e(t) = 0$*, then* $\lim_{t\to\infty} z_i = 0$ *for* $1 \leq i \leq n$*.*

By Lemma 3.1.3, to make the estimate $\hat{x}$ asymptotically converge to x, it is sufficient to choose suitable v and $\hat{f}_k$ such that e converges to zero. We estimate $\hat{f}_k$ as in Section 2.3, where $\hat{f}_k$ is defined in (2.22).

3.1.3 Update Laws

Since we assume that f is unknown, the parameter vector $\theta^*_{f_k}$ is unknown for each k. We update θ_{f_k} using the following adaptive laws.

$$\dot{\hat{\theta}}_{f_k} = \Gamma_{f_k} \bar{\omega}_k e \Phi_k \tag{3.10}$$

where Γ_{f_k} are positive constant matrices. The motivation for this update law is almost the same as that in the last chapter.

3.1.4 Stabilizing Observer Signal

To make the state $\hat{x}$ of the locally weighted learning observer (3.5) asymptotically converge to the state x, the stabilizing observer signal is chosen as

$$v = l_1 z_2 + \cdots + l_{n-1} z_n + Ke + \epsilon_f \text{sign}(e) \tag{3.11}$$

where L is defined in (3.9), K is a positive constant, and

$$\epsilon_f = \begin{cases} \max\{|\epsilon_{f_k}|\}, & \text{if } x \in \mathcal{D}^n \\ h(x), & \text{if } x \in R^n - \mathcal{D}^n. \end{cases} \tag{3.12}$$

Lemma 3.1.4 *For system (3.5), with the stabilizing observer signal v defined in (3.11), locally weighted learning (2.21), update algorithm (3.10), we have*

$$\lim_{t \to \infty} (x - \hat{x}) = 0. \tag{3.13}$$

Proof: By eqn. (3.8), for $x \in R^n - \mathcal{D}^n$ we have

$$\dot{e} = l_1 z_2 + \cdots + l_{n-1} z_n + f(x) - v = -Ke + f(x) - \epsilon_f \text{sign}(e). \tag{3.14}$$

noting the definition of ϵ_f, it is easy to show that e is exponentially converging, which means that $(x - \hat{x})$ is exponentially converging. For $x \in \mathcal{D}^n$ we have

$$\dot{e} = l_1 z_2 + \cdots + l_{n-1} z_n + \sum_k \bar{\omega}_k(x) \overline{x}_k^T \tilde{\theta}_{f_k} + \delta_f(x) - v \tag{3.15}$$

where $\tilde{\theta}_{f_k} = \theta^*_{f_k} - \theta_{f_k}$. Define the positive Lyapunov function as

$$V = \frac{1}{2} e^2 + \sum_k \tilde{\theta}_{f_k}^\top \Gamma_{f_k}^{-1} \tilde{\theta}_{f_k}.$$

Differentiating it along the solution of (3.15), we have

$$\dot{V} = -Ke^2 + e\delta_f - \epsilon_f|e| \leq -Ke^2.$$

So, e asymptotically converges to zero and θ_{f_k} are bounded. By Lemma 3.1.3, we can prove that $\lim_{t\to\infty}(x - \hat{x}) = 0$. ■

Remark 3.1.5 *The approximator parameter μ is a control parameter. It affects the number of local regions (N) and the magnitude of v through ϵ_f. If μ increased, N will decrease but the magnitude of ϵ_f will increase. So, there is a trade-off between the control magnitude and computation burden when we choose μ. Besides LWL observer, other observers can be proposed. The proposed LWL observer has the virtue of simplicity in structure.*

Remark 3.1.6 *To deal with the uncertainty, in this section we proposed a LWL observer. There are many other ways to deal with the uncertainty. One motivation for the presented approach is that with the proposed observer, the pointwise min-norm problem defined later is easily solved.*

With the aid of Lemma 3.1.4, we can design optimal controllers based on system (3.5), i.e., we can design an optimal controller u such that

$$\bar{J}_\infty = \int_0^\infty (q(\hat{x}) + u^2)d\tau \tag{3.16}$$

achieves its minimum. Since $\hat{x}$ asymptotically converges to x, x will converge to zero when $\hat{x}$ converges to zero.

3.1.5 Point-wise Min-norm Controller

With the aid of the locally weighted learning observer, it may seem that the optimal control problem of (3.16) can be solved by using the dynamic programming technique. In fact, the optimal control problem (3.16) is not generally solvable because the dynamics of $\hat{x}$ are nonlinear. To obtain an analytical control law, we consider the pointwise min-norm problem proposed in [24, 56] instead. Before defining the problem, we need some preparation.

A control Lyapunov function (CLF) of system (3.5) is a continuously differentiable, positive definite function $V(y)$: $R^n \to R^+$ such that

$$\inf_u [V_{\hat{x}}\bar{f} + V_{\hat{x}}\bar{g}u] < 0 \tag{3.17}$$

for all $\hat{x} \neq 0$ [69, 70], where

$$\begin{aligned} \bar{f} &= [\hat{x}_2, \ldots, \hat{x}_n, f_0(x) + \hat{f}(x) + v]^\top \\ \bar{g} &= [0, \ldots, 0, g_0(x)]^\top. \end{aligned}$$

If there is a CLF such that eqn. (3.17) is satisfied, the control input u obtained at each point from eqn. (3.17) can make the state of system (3.5) converge to zero. This can be seen when we choose V as a Lyapunov function under those control actions. For a general nonlinear system, it may be difficult to find a CLF or even to determine whether one exists. However, for system (3.5) there exists a CLF. In fact, the function

$$V = \hat{x}^\top P \hat{x} \tag{3.18}$$

is one of the CLFs of system (3.5), where P is a positive definite matrix that satisfies

$$P\Lambda + \Lambda^\top P = -Q$$

for a given positive definite matrix Q,

$$\Lambda = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \alpha_3 & \cdots & \alpha_n \end{bmatrix},$$

and the constants α_i $(1 \leq i \leq n)$ are chosen such that matrix Λ is Hurwitz.

Given a control Lyapunov function V for system (3.5), the pointwise min-norm problem is defined as follows.

Pointwise Min-norm Problem: At each time, select u as

$$\min_u \quad u^2 \tag{3.19}$$

such that

$$V_{\hat{x}}(\bar{f} + \bar{g}u) \leq -\sigma(\hat{x}) \tag{3.20}$$

where σ is a positive definite function of $\hat{x}$ which is chosen according to the trade-off between control effort and stabilizing $\hat{x}$ and such that $V_{\hat{x}}(\bar{f} + \bar{g}u) + \sigma \leq 0$.

Remark 3.1.7 *In the point-wise min-norm problem, V can be any CLF of system (3.5). With different CLF V and σ, different optimal control u can be obtained.*

For the pointwise min-norm problem, we have the following closed form solution which is an extension of the approaches discussed in [55].

Lemma 3.1.8 *For the pointwise min-norm problem (3.19)-(3.20), the optimal control is*

$$u = \begin{cases} -\dfrac{V_{\hat{x}}\bar{f}+\sigma}{V_{\hat{x}}\bar{g}}, & V_{\hat{x}}\bar{g} \neq 0 \\ 0, & V_{\hat{x}}\bar{g} = 0 \end{cases} \tag{3.21}$$

Proof: If $V_{\hat{x}}\bar{g} = 0$, the constraint (3.20) holds automatically by (3.17). So $u = 0$ is the optimal control. If $V_{\hat{x}}\bar{g} \neq 0$, the constraint (3.20) is active. We solve the optimal problem: $\min_u u^2$ such that $V_{\hat{x}}(\bar{f} + \bar{g}u) + \sigma = 0$. By the Lagrange multiplier method, we can obtain $u = -\frac{V_{\hat{x}}\bar{f}+\sigma}{V_{\hat{x}}\bar{g}}$. ■

From Lemma 3.1.8, we can see that the optimal controller u depends on V and σ. Carefully Choosing V and σ may lead to the optimal controller which we are interested in. Sontag's formula [70] provides a choice for σ as

$$\sigma = \sqrt{(V_{\hat{x}}\bar{f})^2 + q(\hat{x})(V_{\hat{x}}\bar{g}\bar{g}^\top V_{\hat{x}}^\top)}. \tag{3.22}$$

It should be noted that if the control Lyapunov function V is chosen as a value function of the HJB equation corresponding to the cost function (3.16) and σ is chosen as (3.22), the optimal control (3.21) would be the solution to the optimal control problem (3.16). This fact lead us to choose σ as in (3.22). In [24], it was shown that every CLF is the value function of some meaningful cost function which means that it solves the HJB equation associated with a meaningful cost. This is referred to "inverse optimal control".

Combining the results in this subsection and the last subsection, we have the following result.

Theorem 3.1.9 *For system (3.1) with the locally weighted learning observer defined in (3.5) with update law (3.10) and stabilizing observer signal (3.11), the optimal control (3.21)* *solves the point-wise min-norm problem (3.19)-(3.20) with a given CLF $V(\hat{x})$ and a suitable positive definite function σ that makes x converge to zero.*

Proof: By Lemma 3.1.8, the optimal control (3.21) solves the pointwise min-norm problem (3.19)-(3.20). Choose V in (3.20) as a Lyapunov function, $\hat{x}$ converges to zero since eqn. (3.20) holds for every point $\hat{x}$. By Lemma 3.1.4, x also converges to zero. ■

In Theorem 3.1.9, there are several control parameters. Constant μ determines the number of the local regions and the magnitude of the control input as time tends to infinity (see Remark 3.1.5). The matrix P in control Lyapunov function V is an important control parameter. By suitably choosing V the performance of the closed-loop system with the controller (3.21) will be close to the performance of the closed-loop system with the optimal controller of the optimal control problem (3.16). Unfortunately, there is no prior knowledge of how to choose CLF V. In practice, one can choose a control Lyapunov function V as in (3.18).

3.1.6 Numerical Example

We consider for illustrative purpose a second order system given by

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \sin(0.4(x_1 + x_2)) + 2x_1 + 3x_2 + 0.2\sin x_1 + \Big(2 + \sin(0.4(x_1 + x_2))\Big)u. \end{aligned}$$

For the example, $x \in \Re^2, u \in \Re$ and we assume that there is only partial priori knowledge of the system nonlinearity. The known 'design models' are $f_0(x_1, x_2) = \sin(0.4(x_1 + x_2))$ and $g_0(x_1, x_2) = 2 + \sin(0.4(x_1 + x_2)$; therefore, the unknown design model error is $f(x) =$

$2x_1 + 3x_2 + 0.2 \sin x_1$. We also assume that the system is designed to operate over the region $(x_1, x_2) \in \mathcal{D}^2 = [-5, 5] \times [-5, 5]$.

Following the results in the previous sections, the locally weighted learning observer is

$$\dot{\hat{x}}_1 = \hat{x}_2 \tag{3.23}$$

$$\dot{\hat{x}}_2 = f_0(x) + \hat{f} + g(x)u + v \tag{3.24}$$

where $\hat{f}$ is an approximation to f with locally weighted learning algorithm (3.10) and v is defined in (3.11). In the locally weighted learning observer, we choose $\mu = 0.5$, $\lambda = 1$. The function approximation accuracies are specified as $\epsilon_f = 0.05$.

The weighting function is the biquadratic kernel of the form as

$$\omega_k(x) = \begin{cases} \left(1 - R^2\right)^2, & \text{if } R < 1 \\ 0, & \text{otherwise.} \end{cases} \tag{3.25}$$

where

$$R = \left\| \frac{|x_1 - c_{k,1}|}{\mu}, \frac{|x_2 - c_{k,2}|}{\mu} \right\|_2 . \tag{3.26}$$

The local basis function is $\bar{x}_k = [1, x_1 - c_{k,1}, x_2 - c_{k,2}]^\top$ with c_k being the center of the $\bar{S}_k$. Therefore, each f_k is the optimal local affine approximation to f on $\bar{S}_k$. We select $c_{k,1} = c_{k,2} = \frac{k\mu}{2}$ for $-20 \leq k \leq 20$. For simplicity, we choose the initial conditions of $\theta_{f_k}(0) = 0$ The adaptation rate matrices are set to $\Gamma_{f_k} = \text{diag}([1, 1, 1])$ where $\text{diag}(v)$ is the square diagonal matrix with diagonal component equal to the vector v. We choose $Q = I_{2\times 2}$. The Lyapunov function V is obtained from (3.18). The function σ is chosen as in (3.22). So, the control law is as in (3.21).

Figs. 3.1 and Fig. 3.2 show the responses of $\hat{x}$ and x. It can be seen that $\hat{x}$ and x converge to zero. Fig. 3.3 shows the response of the intermediate variable e, which converges to zero. The control input converges to zero as x converges to zero. The responses of θ_{f_k} are not shown here. However, they are bounded.

If there is no learning, i.e. $\hat{f} = 0$, the control law cannot make the state converge to zero. Fig. 3.4 shows the response of the state without learning. It can be seen that x does not converge to zero.

3.2 Pointwise Min-norm Optimal Tracking Control

In this section, we consider the optimal tracking control of the uncertain nonlinear system shown in (3.1).

3.2.1 Problem Statement

Given a desired bounded trajectory $x^d = [x_1^d, x_2^d, \ldots, x_n^d]^\top$ which satisfies

$$\dot{x}_1^d = x_2^d, \quad \dot{x}_2^d = x_3^d, \ldots, \quad \dot{x}_{n-1}^d = x_n^d. \tag{3.27}$$

The problem discussed in this section is to design an optimal controller u such that the cost function

$$J_\infty = \int_0^\infty [(x - x^d)^\top Q(x - x^d) + u^2] d\tau \tag{3.28}$$

achieves its minimum, where Q is a positive definite matrix.

If $f(x)$ is known, a standard dynamic programming argument reduces the above optimal control problem to finding the value function V^* solving the Hamilton-Jacobi-

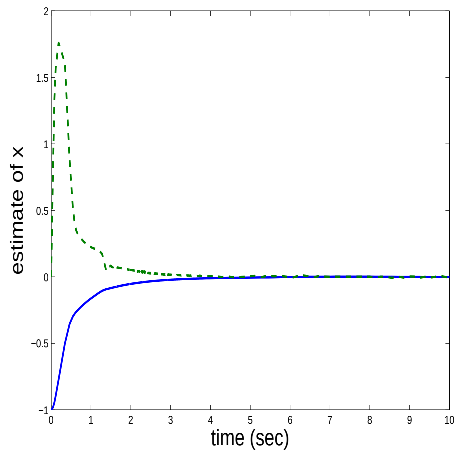

Figure 3.1: Response of $\hat{x}$ ($\hat{x}_1$: solid, $\hat{x}_2$: dashed)

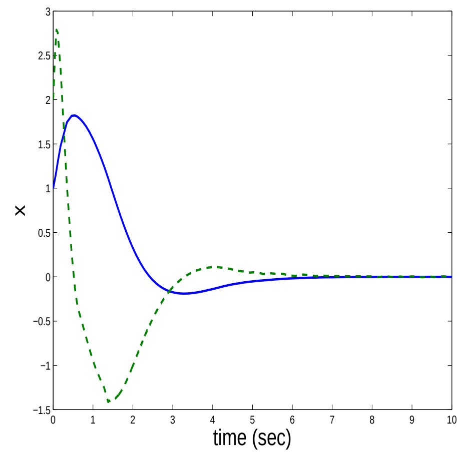

Figure 3.2: Response of x (x_1: solid, x_2: dashed)

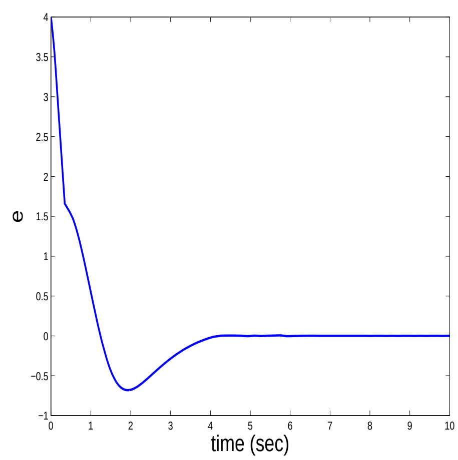

Figure 3.3: Response of e

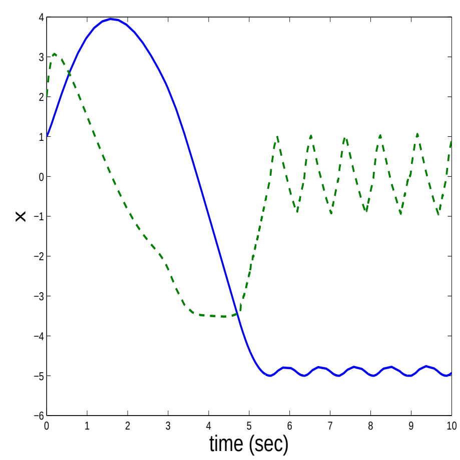

Figure 3.4: Response of x without learning (x_1: solid, x_2: dashed)

Bellman partial differential equation (HJB)

$$V_x^* f_e - \frac{1}{4}\left(V_x^* g_e g_e^\top V_x^{*\top}\right) + (x - x^d)^\top Q(x - x^d) = 0 \tag{3.29}$$

where V_x^* denotes $\frac{\partial V^*}{\partial x}$, $f_e = [x_2, x_3, \ldots, x_n, f_0(x) + f(x)]^\top$, and $g_e = [0, \ldots, 0, g_0(x)]^\top$. If there exists a continuously differentiable positive definite solution $V^*(x)$ to eqn. (3.29), then the optimal controller is

$$u = -\frac{1}{2} g_e^\top V_x^{*\top}. \tag{3.30}$$

Since $f(x)$ is unknown, we propose a locally weighted learning observer (LWLO) to estimate $f(x)$ as discussed in the last section. Then, we solve a modified optimal control problem.

3.2.2 Locally Weighted Learning Observer

Let the observer be defined as in (3.5), where $\hat{x} = [\hat{x}_1, \ldots, \hat{x}_n]^\top$ is the estimate of x, v is a stabilizing observer signal, $\hat{f}(x)$ is the estimate of $f(x)$ based on a locally weighted learning (LWL) algorithm defined in (2.21) with adaptive laws (3.10). Define

$$z = [z_1, \ldots, z_n]^\top = x - \hat{x} \tag{3.31}$$

$$e = L^\top z \tag{3.32}$$

where L is defined in (3.9). We have Lemma 3.1.3 which means to make the estimate $\hat{x}$ asymptotically converge to x, it is sufficient to choose suitable v and $\hat{f}_k$ such that e converges to zero.

To make the state of the locally weighted learning observer (3.5) asymptotically converge to the state x, the stabilizing observer signal is chosen as

$$v = l_1 z_2 + \cdots + l_{n-1} z_n + Ke + \frac{\epsilon_f e}{\sqrt{e^2 + e^{-t}}} \tag{3.33}$$

where K is a positive constant, ϵ_f is a constant and is defined in (3.12). The introduction of the term e^{-t} in (3.12) is to make v differentiable.

Lemma 3.2.1 *For system (3.5), with the stabilizing observer signal v defined in (3.33), locally weighted learning (2.21), update algorithm (3.10), then $(x - \hat{x})$ converges to zero and $\hat{\theta}_{f_k}$ are bounded.*

Proof: By eqn. (3.32), we have

$$\dot{e} \;=\; l_1 z_2 + \cdots + l_{n-1} z_n + \sum_k \bar{\omega}_k(x) \bar{x}_k^T \tilde{\theta}_{f_k} + \delta_f(x) - v \tag{3.34}$$

where $\tilde{\theta}_{f_k} = \theta_{f_k}^* - \theta_{f_k}$. Define the positive Lyapunov function

$$V_1 = \frac{1}{2} e^2 + \sum_k \tilde{\theta}_{f_k}^\top \Gamma_{f_k}^{-1} \tilde{\theta}_{f_k}.$$

Differentiating it along the solution of (3.34), we have

$$\dot{V}_1 \;=\; -Ke^2 + e\delta_f - \frac{\epsilon_f e^2}{\sqrt{e^2 + \exp(-t)}} \leq -Ke^2 + \epsilon_f \exp(-t/2).$$

Therefore, V_1 is bounded by integrating both sides, which means that θ_{f_k} and e are bounded. By integrating both sides, it can be shown that e^2 is integrable. Therefore, e converges to zero. By Lemma 3.1.3, we can prove that $(x - \hat{x})$ converges to zero. ■

Remark 3.2.2 *In the observer, we apply the locally weighted learning idea. The advantages of the locally weighted learning are two fold. First of all, the approximation errors are functions of local approximators. Secondly, the burden of the computation for learning is relieved.*

Remark 3.2.3 *In the observer, μ is a control parameter. It affects the number of local regions (N) and the magnitude of v through ϵ_f. If μ is large, in general N will be small but the magnitude of the last term in v may be large. Alternatively, as N is increased, the magnitude of the last term in v will decrease. So, the choice of μ involves a trade-off between the control magnitude and computation burden.*

With the aid of Lemma 3.2.1, we can design optimal controllers for system (3.5) theoretically, i.e., we can design an optimal controller u such that

$$\bar{J}_\infty = \int_0^\infty \left((\hat{x} - x^d)^\top Q(\hat{x} - x^d) + u^2 \right) d\tau \tag{3.35}$$

achieves its minimum. Since $\hat{x}$ is close to x as time converges to infinity, x will converge to a small neighborhood of x^d if $\hat{x}$ converges to a small neighborhood of x^d.

3.2.3 Point-wise Min-norm Controller

In practice, the optimal control problem (3.35) is not generally solvable because the dynamics of $\hat{x}$ is nonlinear. To obtain an analytical control law, we consider the point-wise min-norm problem as in Section 3.2.2. Let $q = [q_1, \ldots, q_n]^\top$ and

$$q_i = \hat{x}_i - x_i^d, \quad 1 \le i \le n,$$

then

$$\begin{aligned} \dot{q}_i &= q_{i+1}, \quad 1 \le i \le n-1 \\ \dot{q}_n &= f_0(x) + \hat{f}(x) + g_0(x)u + v - \dot{x}_n^d. \end{aligned} \tag{3.36}$$

Given a control Lyapunov function $V(q)$ for system (3.36), the point-wise min-norm problem is defined as follows.

Point-wise Min-norm Problem:

$$\min_u \quad u^2 \tag{3.37}$$

such that

$$V_q(\bar{f} + \bar{g}u) \leq -\sigma(q) \tag{3.38}$$

where σ is a positive definite function of q which is chosen by the designer.

For the point-wise min-norm problem, we have the following closed form solution.

Lemma 3.2.4 *For the point-wise min-norm problem (3.37)-(3.38), the optimal control is*

$$u = \begin{cases} -\dfrac{V_q\bar{f} + \sigma}{V_q\bar{g}}, & V_q\bar{g} \neq 0 \\ 0, & V_q\bar{g} = 0 \end{cases} \tag{3.39}$$

Proof: The proof is the same as the proof of Lemma 3.1.8. ■

An example min-norm optimal controller is given by Sontag's formula [70] as follows.

Lemma 3.2.5 *For the point-wise min-norm problem (3.37)-(3.38), if*

$$\sigma = \sqrt{(V_q\bar{f})^2 + q^\top Qq(V_q\bar{g}\bar{g}^\top V_q^\top)} \tag{3.40}$$

then the optimal control law is

$$u = \begin{cases} -\left[\frac{V_q\bar{f} + \sqrt{(V_q\bar{f})^2 + q^\top Qq(V_q\bar{g}\bar{g}^\top V_q^\top)}}{V_q\bar{g}\bar{g}^\top V_q^\top}\right]\bar{g}^\top V_q^\top, & V_q\bar{g} \neq 0 \\ 0, & V_q\bar{g} = 0 \end{cases} \tag{3.41}$$

Combining the results in this subsection and Subsection 3.2.2, we have the following result.

Theorem 3.2.6 *For system (3.1) with the locally weighted learning observer defined in (3.5) with update laws (3.10) and stabilizing observer signal (3.33), the optimal control (3.41) solves the point-wise min-norm problem (3.37)-(3.38) with a given CLF $V(q)$ and make $(x - x^d)$ converges to zero.*

Proof: The proof is almost the same as the proof of Theorem 3.1.9. ■

3.2.4 Numerical Example

We consider for illustrative purpose a second order system given by

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= \sin(0.4(x_1 + x_2)) + \Big(2 + \sin(0.4(x_1 + x_2))\Big)u.\end{aligned}$$

For the example, $x \in \Re^2, u \in \Re$ and we assume that there is only partial priori knowledge of the system nonlinearities. The known 'design model' has $f_0(x_1, x_2) = 0.4(x_1 + x_2)$ and $g_0(x_1, x_2) = 2 + \sin(0.4(x_1 + x_2)$; therefore, the unknown design model error is $f(x) = \sin(0.4(x_1 + x_2)) - 0.4(x_1 + x_2)$.

Given a desired trajectory x^d, we want to design an optimal control in the sense of point-wise min-norm such that $(x - x^d)$ converges to zero.

The locally weighted learning observer is (3.23)-(3.24) where $\hat{f}$ is an online approximation to f with locally weighted learning algorithm (3.10) and v is defined in (3.33). In the locally weighted learning observer, we choose $\mu = 0.5$, $\lambda = 1$. The function approximation accuracies are specified as $\epsilon_f = 0.03$.

The weighting function is the biquadratic kernel of the form as (3.25) where R is defined in (3.26). The local basis function is $\bar{x}_k = [1, x_1 - c_{k,1}, x_2 - c_{k,2}]^\top$ with c_k being the center of the $\bar{S}_k$. Therefore, f_k is the optimal local affine approximation to f on $\bar{S}_k$. We select $c_{k,1} = c_{k,2} = \frac{k\mu}{2}$ for $-20 \leq k \leq 20$. For simplicity, we choose the initial conditions of $\theta_{f_k}(0) = 0$ The adaptation rate matrices are set to $\Gamma_{f_k} = \text{diag}([1,1,1])$ where $\text{diag}(v)$ is the square diagonal matrix with diagonal component equal to the vector v.

For $x^d = [\sin t, \cos t]^\top$, Fig. 3.5 and Fig. 3.6 show the responses of $\hat{x}$ and x with/without learning. Fig. 3.7 shows the tracking errors $x - x^d$ with and without learning.

3.3 Conclusion

This chapter considers the optimal regulation and optimal tracking control of uncertain nonlinear systems. Optimal regulation and optimal tracking controllers are proposed for the observers in the sense of point-wise min-norm, respectively. The advantage of the proposed methods is that analytic optimal controllers are proposed and the stability of the closed-loop system is guaranteed. Furthermore, if the control Lyapunov functions V are the value functions of the HJB equations corresponding to the cost functions (3.16) and (3.35) when σ are chosen as (3.22) and (3.40). The optimal regulation controller (3.21) and the optimal tracking controller (3.41) will be the solutions to the optimal regulation control problem (3.16) and the optimal tracking control problem (3.35), respectively.

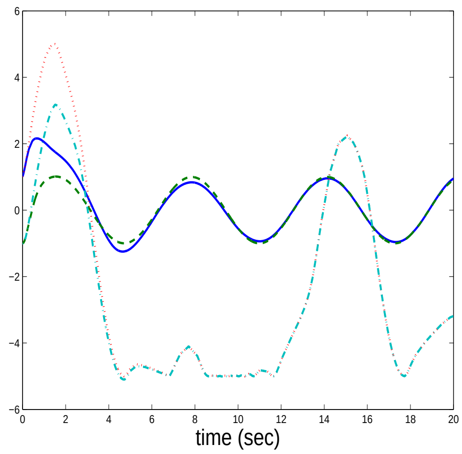

Figure 3.5: Response of $\hat{x}_1$ and x_1 with and without learning (x_1 with learning: solid, $\hat{x}_1$ with learning: dashed, x_1 without learning: dotted, $\hat{x}_1$ without learning: dashdot)

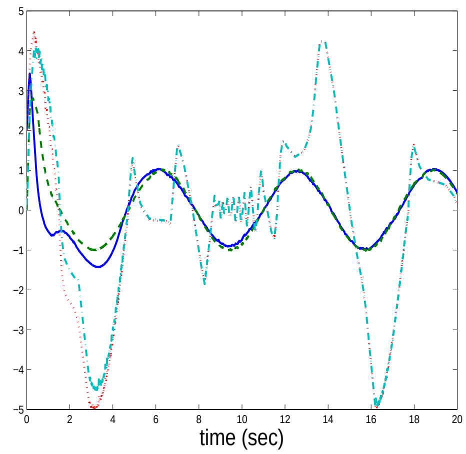

Figure 3.6: Response of $\hat{x}_2$ and x_2 with and without learning (x_2 with learning: solid, $\hat{x}_2$ with learning: dashed, x_2 without learning: dotted, $\hat{x}_2$ without learning: dashdot)

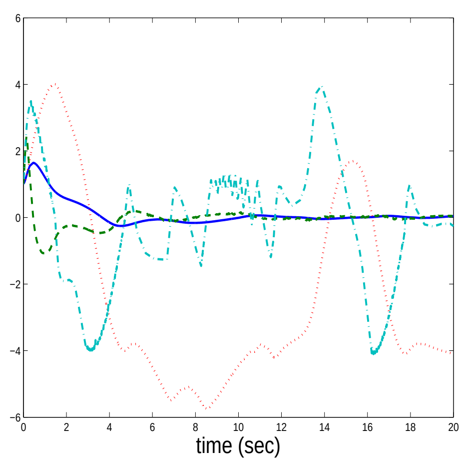

Figure 3.7: Response of the tracking errors $x - x^d$ with and without learning ($x_1 - x_1^d$ with learning: solid, $x_2 - x_2^d$ with learning: dashed, $x_1 - x_1^d$ without learning: dotted, $x_2 - x_2^d$ without learning: dashdot)

Chapter 4

Approximately Optimal Control of Uncertain Nonlinear Systems

In this chapter, we consider the optimal control of the uncertain nonlinear system shown in (4.1). Adaptive approximately optimal controllers are proposed by considering the ideas in adaptive critic design and the approximately optimal control algorithms in [6, 62, 73]. In the proposed algorithms the controllers are updated according to the information of the value function.

This chapter is organized as follows. Section 4.1 defines the problem. Section 4.2 contains some preliminary results in approximately optimal control. The approximately optimal controllers are presented in Sections 4.3-4.4. The effectiveness of the proposed controllers is shown in Section 4.5. The last section concludes this chapter.

4.1 Problem Statement

Consider an n-th order nonlinear system

$$\dot{x}_i = x_{i+1}, \quad 1 \le i \le n-1 \tag{4.1}$$

$$\dot{x}_n = f(x) + g(x)u \tag{4.2}$$

where $x = [x_1 \ldots, x_n]^\top \in R^n$ is the state, and $u \in R$ is the input. We assume that the system is Lipschitz continuous on a compact set $\mathcal{D}^n \subset R^n$ that contains the origin and that the system is stabilizable on $\mathcal{D}^n$. During the control, we assume $f(x)$ is unknown for $x \in \mathcal{D}^n$, while $g(x)$ is known.

The problem discussed in this section is to design an optimal controller u such that the cost function (3.2) There are two obstacles which prevent us from finding the optimal controller. The first one is that $f(x)$ is unknown. The second one is that it is extremely difficult to solve the HJB partial differential equation even if $f(x)$ is known. To overcome the two obstacles, in the last chapter we proposed an locally weighted observer and defined a point-wise min-norm problem. In this chapter, to overcome the first obstacle, we introduce a dynamic equation to estimate the value function so that the information about $f(x)$ is not directly required in the controller as in [73]. To deal with the second obstacle, we use learning algorithms to approximate the value function. To propose a control law, we make the following additional assumptions.

Assumption 4.1.1 *The unknown function $f(x)$ satisfies $|f(x)| \le \xi(x)$ for $x \in R^n - \mathcal{D}^n$, where $\xi(x)$ is a known function.*

Assumption 4.1.2 *The function $g(x) \neq 0$ for any x.*

Assumption 4.1.1 is useful for controller design when x is outside $\mathcal{D}^n$. For $x \in R^n - \mathcal{D}^n$, we design a slide mode controller such that x enters $\mathcal{D}^n$ quickly [68]. Once $x \in \mathcal{D}^n$, we design an adaptive approximately optimal controller based on the cost function (3.2). The focus of this chapter is on how to design adaptive approximately optimal controllers for $x \in \mathcal{D}^n$.

4.2 An Approximation Theory for Optimal Regulation

Since it is hard to obtain an optimal controller by solving HJB partial differential equations, many methods have been proposed to approximate the optimal controller. For example, approximate dynamic programming, model predictive control, etc. For optimal control with finite horizon, approximately optimal controllers were proposed in paper [62] with offline calculation. Next, we give some results from [62, 73, 74] on how to approximate the optimal controller under the condition that $f(x)$ is known.

Define the pre-Hamiltonian function for some control law u and an associated $V(x,t;u_1)$:

$$H\left(x, \frac{\partial V}{\partial x}, u, t\right) = x^\top Q x + u^2 + \frac{\partial V^\top}{\partial x} F(x) + \frac{\partial V^\top}{\partial x} G(x) u \tag{4.3}$$

where

$$F(x) = \begin{bmatrix} x_2 \\ \vdots \\ x_n \\ f(x) \end{bmatrix}, \quad G(x) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ g(x) \end{bmatrix}.$$

Lemma 4.2.1 ([62]) *Assume u_1 is an asymptotically stabilizing controller of (4.1). If* *there exists a positive definite continuously differentiable function $V(x,t;u_1)$ satisfying the following properties*

$$\frac{\partial V}{\partial t} + \frac{\partial V^\top}{\partial x} F(x) + \frac{\partial V^\top}{\partial x} G(x) u_1 + x^\top Q x + u_1^2 = 0 \tag{4.4}$$

$$V(0,\infty;u_1) = 0 \tag{4.5}$$

then $V(x,t;u_1)$ is the value function of the system (4.1) for all t, and

$$V(x(t_0),t_0;u_1) = J(u_1;x(t_0),t_0) = \int_{t_0}^{\infty} (x^\top Q x + u_1^2) d\tau. \tag{4.6}$$

Corollary 4.2.2 ([62]) *The optimal control law u^* and the optimal value function $V^*(x,t;u^*)$, if they exist, satisfy Lemma 4.2.1 and*

$$0 < V^*(x,t;u^*) \leq V(x,t;u), \quad u^* \neq u. \tag{4.7}$$

Lemma 4.2.3 ([62]) *If a sequence of pairs $\{u_i, V_i\}$ satisfying Lemma 4.2.1 is generated by selecting the control u_i to minimize the pre-Hamiltonian function (4.3) with the previous value function V_{i-1}, e.g.,*

$$u_{i+1} = -\frac{1}{2} G^\top \frac{\partial V_i}{\partial x}, \tag{4.8}$$

then the corresponding value function V_i satisfies the inequality

$$V_i \geq V_{i+1}.$$

Moreover, $\lim_{i\to\infty} V_i = V^*(x,t;u^*)$ where V^* is the cost function corresponding to the optimal cost.

Lemma 4.2.3 provides a systematic method to calculate the optimal controller for system (4.1) with known $f(x)$ and $g(x)$. In [62], two procedures for design of approximately optimal controllers were presented, e.g., exact design procedure and approximate design procedure. In each procedure, it needs to find analytic expression of V_i. However, for a general nonlinear system it is hard to find the solution V from equation (4.5) because it is a partial differential equation. Therefore, it is hard to find the optimal controller or a nearly optimal controller for system (4.1) with the aid of the results in Lemma 4.2.1 and Lemma 4.2.3. In this paper, we assume $f(x)$ is unknown. So, it is impossible to solve the partial differential equations in each step with unknown $f(x)$. However, we can obtain a numerical solution for a period of time with the aid of Lemma 4.2.1. In fact, by Lemma 4.2.1,

$$V_i(x(t), t; u_i) = \int_t^{\infty} (x^\top Qx + u_i^2) d\tau.$$

Then

$$V_i(x(t), t; u_i) = \int_t^{t+T} (x^\top Qx + u_i^2) d\tau + V_i(x(t+T), t+T; u_i), \quad V_i(0, \infty; u_i) = 0. \tag{4.9}$$

Therefore, if u_i is known, the numerical solution of $V_i(x(t), t; u_i)$ can be obtained by solving (4.9) for a given time interval T [73, 74]. If T is small, V_i obtained from (4.9) is close to the optimal value function. It should be noted that in (4.9) $f(x)$ does not appear. Which means $f(x)$ is not necessary for solving V_i.

In order to calculate u_{i+1} with the aid of Lemma 4.2.3, it needs an analytic form of $V_i(x(t), t; u_i$. However, by equation (4.9) we can only get a numerical solution of $V_i(x(t), t; u_i$. To overcome it, we assume V_i is in a special form and is updated by (4.9). With the aid of this known form, we can calculate u_{i+1}.

4.3 An Adaptive Approximately Optimal Controller

We assume V_i has a special form as follows

$$V_i = \sum_{j=1}^{m} \phi_j(x)\theta_{ji} + \epsilon_i(x) = \Phi^\top(x)\theta_i + \epsilon_i(x) \tag{4.10}$$

where $\phi_j(x)$ is a known basis function, θ_{ji} is unknown constant parameter, ϵ_i is the approximation error, $\Phi(x) = [\phi_1(x), \ldots, \phi_m(x)]^\top$ and $\theta_i = [\theta_{1i}, \ldots, \theta_{mi}]^\top$. By the well-known approximation results in neural networks, for a compact domain $\mathcal{D}$ and a given $\mu > 0$, there exists a sufficient large number m such that $|\epsilon_i(x)| < \mu$ [14].

Substitute (4.10) into (4.9), we have

$$\int_t^{t+T} (x^\top Q x + u_i^2) d\tau = [\Phi(x(t)) - \Phi(x(t+T))]^\top \theta_i + \bar{\epsilon}_i(t) \tag{4.11}$$

or in short

$$y_i(t) = \alpha_i^\top(t)\theta_{*i} + \bar{\epsilon}_i(t) \tag{4.12}$$

where

$$\begin{aligned} y_i(t) &= \int_t^{t+T} (x^\top Q x + u_i^2) d\tau \\ \alpha_i(t) &= \Phi(x(t)) - \Phi(x(t+T)) \\ \bar{\epsilon}_i(t) &= \epsilon_i(x(t)) - \epsilon_i(x(t+T)). \end{aligned}$$

Let $\hat{\theta}_i$ be an estimate of θ_i, then

$$y_i = \alpha_i^\top \hat{\theta}_i + e_i \tag{4.13}$$

where e_i is the approximation error.

To derive an adaptation law, we minimize the individual weighted squared error e_i,

$$\bar{J}_i(\theta) = \int_{\Omega_i} \left(y_i - \alpha^\top \hat{\theta}_i \right)^2 d\tau \tag{4.14}$$

where Ω_i denotes the time interval over which control u_i is applying to the system. Let

$$\frac{\partial \bar{J}_i}{\partial \hat{\theta}_{\ i}} = 0,$$

we obtain

$$\int_{\Omega_i} \alpha_i y_i \ d\tau = \int_{\Omega_i} \alpha_i \alpha_i^\top \ d\tau \hat{\theta}_i \tag{4.15}$$

for $y_i \in \Re$. Theoretically, it can be derived that

$$\hat{\theta}_i = \left(\int_{\Omega_i} \alpha_i \alpha_i^\top dt \right)^{-1} \int_{\Omega_i} \alpha_i y_i dt = \Gamma_i^{-1} \int_{\Omega_i} \alpha_i y_i dt \tag{4.16}$$

where

$$\Gamma_i = \int_{\Omega_i} \alpha_i \alpha_i^\top dt. \tag{4.17}$$

In (4.16), Γ_i should be invertible. In [73], Γ_i is assumed to be non-singular everywhere and choose the following controller.

$$u_{i+1} = -\frac{1}{2} G^\top \left[\frac{\partial \Phi}{\partial x} \right]^\top \hat{\theta}_i. \tag{4.18}$$

In (4.18), in order to calculate $\hat{\theta}_i$ it needs to calculate the inverse of Γ_i. It is hard to make Γ_i non-singular everywhere. Also, if there are many basis functions, the dimension of Γ_i may be large. The computation of the inverse of Γ_i needs lots of resources. To overcome this problem, we derive another version of adaptive approximately optimal controller. Let

$$P^{-1}(t) = \int_0^t \alpha_i \alpha_i^\top \ d\tau. \tag{4.19}$$

Taking the time derivative of (4.19), we obtain the derivative of $P^{-1}(t)$ as

$$\frac{d}{dt}P^{-1}(t) = \alpha_i(t)\alpha_i^\top(t). \tag{4.20}$$

Note that the identity $P(t)P^{-1}(t) = I$ yields

$$\frac{d}{dt}P(t)P^{-1}(t) = \dot{P}(t)P^{-1}(t) + P(t)\dot{P}^{-1}(t) = 0$$

which implies that

$$\dot{P}(t) = -P(t)\dot{P}^{-1}(t)P(t).$$

After substitution of (4.20), we can write the update law for $P(t)$ as

$$\dot{P}(t) = -P(t)\alpha_i(t)\alpha_i^\top(t)P(t). \tag{4.21}$$

Finally, taking the time derivative on both sides of (4.15), we solve for $\dot{\hat{\theta}}_i$ as

$$\alpha_i y_i = (\alpha_i\alpha_i^\top)\hat{\theta}_i + P^{-1}\dot{\hat{\theta}}_i$$

which yields the value function parameter update based on the prediction error $(y_i - \alpha_i^\top\hat{\theta}_i)$ as

$$\dot{\hat{\theta}}_i(t) = P(t)\left[\alpha_i(t)(y_i(t) - \alpha_i^\top(t)\hat{\theta}_i(t))\right]. \tag{4.22}$$

According to (4.8), we choose the control law

$$u = \begin{cases} -\frac{1}{2}g^\top\left[\frac{\partial\Phi}{\partial x}\right]^\top\hat{\theta}_i, & \text{if } x \in \mathcal{D}^n \\ g^{-1}\left[-ke - \sum_{s=1}^{n-1} l_s x_{s+1} - \xi\text{sign}(e)\right], & \text{if } x \notin \mathcal{D}^n, \end{cases} \tag{4.23}$$

where $k > 0$,

$$e = L^\top x = \sum_{s=1}^{n} l_s x_s \tag{4.24}$$

$$L = [l_1, l_2, \ldots, l_{n-1}, l_n]^\top = [\lambda^{n-1}, C_{n-1}^1\lambda^{n-2}, \ldots, C_{n-1}^{n-2}\lambda, 1]^\top, \quad \lambda > 0. \tag{4.25}$$

We summarize the above results as follows.

Control Algorithm 4.1: For system (4.1), an approximately optimal control law is (4.23) with adaptive law (4.22) and (4.21).

Remark 4.3.1 *In Control Algorithm 4.1, for* $x \in R^n - \mathcal{D}^n$ *the controller is a sliding mode control. It will make the state of the system come into* $\mathcal{D}^n$ *in finite time. For* $x \in \mathcal{D}^n$*, the controller is adaptive. The update laws (4.21)-(4.22) play the role of estimating* V_i *iteratively. With the aid of the special structure of the value function, the value function at each step is approximated with* $\hat{\theta}$ *updated online. Therefore, with the aid of the learning algorithm, the solutions to partial differential equations can be obtained. Fig. 4.1 shows the diagram of the adaptive approximately optimal control, where* y_i*,* α_i*, and* $\hat{\theta}_{*i}$ *are replaced with* y*,* α*, and* $\hat{\theta}$ *for convenience.*

Remark 4.3.2 *To calculate* V_i *numerically,* T *should be chosen carefully. Small* T *is good for the closed-loop system performance. However, if* T *is too small other problems may arise. In the adaptive approximately optimal control,* T *can be considered as time delay (see Fig. 4.1).*

Remark 4.3.3 *In [73],* V_i *is approximated by a neural network. The weights are calculated by standard least square. To solve the weights, a non-singularity condition is required on the matrix* Γ_i*, which is difficult to satisfy. In this paper, the unknown parameters are updated by adaptive laws. No non-singularity condition is not required for adaptation of the parameters.*

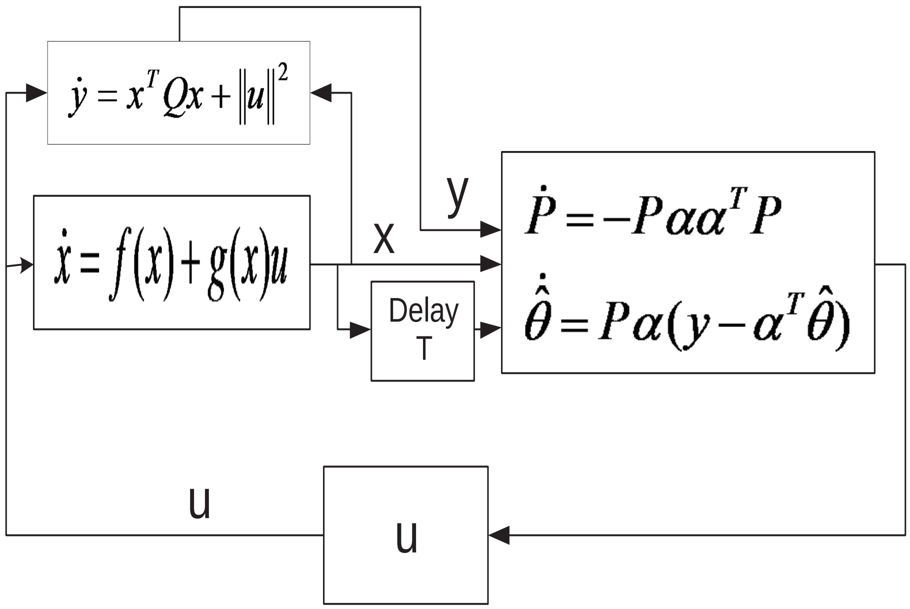

Figure 4.1: Diagram of the adaptive approximately optimal control

4.4 Locally Weighted Learning Optimal Controller

In the last section, the value function V_i is approximated by a set of base functions for $x \in \mathcal{D}^n$. In this section, we apply locally weighted learning (LWL) techniques to approximate the value function V_i.

In locally weighted learning, the approximation of $h(x)$ ($h(x) = V_i$ for notation simplicity) at a point x are formed from the normalized weighted average of local approximators $\hat{h}_k(x)$ such that

$$\hat{h}(x) = \frac{\sum_k \omega_k(x)\hat{h}_k(x)}{\sum_l \omega_l(x)} \tag{4.26}$$

where each ω_k is nonzero only on a set denoted by S_k and

$$S_k = \Big\{ x \in \mathcal{D}^n \mid \omega_k(x) \neq 0 \Big\}. \tag{4.27}$$

ω_k is chosen such that

$$\mathcal{D}^n = \bigcup_{1 \le k \le N} S_k$$

and

$$S_k \cap S_j = \emptyset, \quad \forall k \neq j$$

for convenience. We choose the function $\hat{h}_k$ to be an approximation of $h(x)$ for $x \in S_k$ that is defined as

$$\hat{h}_k(x) = \Phi^\top(x)\hat{\theta}_k \tag{4.28}$$

where $\Phi(x)$ is a base function vector for $x \in S_k$. Since S_k is a subset of $\mathcal{D}^n$, the number of the base functions in $\Phi(x)$ is smaller than that in the last subsection. Let

$$\bar{\omega}_k(x) = \frac{\omega_k(x)}{\sum_{k=1}^N \omega_k(x)}.$$

For any $x \in \mathcal{D}^n$, $h(x)$ can be represented as the weighted sum of the local approximators:

$$h(x) = \sum_k \bar{\omega}_k(x) h_k(x) + \delta(x) = \sum_k \bar{\omega}_k(x)\Phi^\top(x)\theta_k + \delta(x) \tag{4.29}$$

where $\delta(x)$ is the approximation error. For $x \in \bar{S}_k$ where $\bar{S}_k$ denotes the closure of S_k,

$$h(x) = \bar{\omega}_k \Phi^\top(x)\theta_k + \epsilon_k(x). \tag{4.30}$$

Since $V_i(x, u_i) = h(x)$, substitute (4.30) into (4.9), we have

$$\begin{aligned} \int_t^{t+T} (x^\top Q x + \|u_i\|^2) d\tau = [\bar{\omega}_k(x(t))\Phi(x(t)) \\ -\bar{\omega}_k(x(t+T))\Phi(x(t+T))]^\top \theta_k + e_k(t+T) \end{aligned} \tag{4.31}$$

or in short

$$y(t+T) = \alpha_k^\top(t+T)\theta_k + \bar{\epsilon}_k(t+T) \tag{4.32}$$

where

$$
\begin{aligned}
y(t+T) &= \int_0^{t+T} (x^\top Q x + \|u_i\|^2) d\tau \\
\alpha_k(t+T) &= \bar{\omega}_k(x(t))\Phi(x(t)) - \bar{\omega}_k(x(t+T))\Phi(x(t+T)) \\
\bar{\epsilon}_k(t+T) &= \epsilon_k(x(t)) - \epsilon_k(x(t+T)).
\end{aligned}
$$

Let $\hat{\theta}_k$ be an estimate of θ_k, then

$$
y = \alpha_k^\top \hat{\theta}_k + e_k \tag{4.33}
$$

where e_k is the approximation error.

To derive the LWL parameter adaptation laws, we minimize the individual weighted squared error e_k for each local model,

$$
\bar{J}_k(\theta_k) = \int_0^t e^{-\int_\tau^t \lambda \omega_k(x(r))dr} \omega_k \left(y - \alpha_k^\top \hat{\theta}_k\right)^2 d\tau. \tag{4.34}
$$

Let

$$
\frac{\partial \bar{J}_k}{\partial \hat{\theta}_k} = 0,
$$

we obtain

$$
\int_0^t e^{-\int_\tau^t \lambda \omega_k(x(r))dr} \omega_k \alpha_k y \ d\tau = \int_0^t e^{-\int_\tau^t \lambda \omega_k(x(r))dr} \omega_k \alpha_k \alpha_k^\top \ d\tau \hat{\theta}_k. \tag{4.35}
$$

Define

$$
P_k^{-1}(t) = \int_0^t e^{-\int_\tau^t \lambda \omega_k(x(r))dr} \omega_k \alpha_k \alpha_k^\top \ d\tau. \tag{4.36}
$$

Taking the time derivative of (4.36), we obtain the derivative of $P_k^{-1}(t)$ as

$$
\frac{d}{dt} P_k^{-1} = \omega_k \alpha_k \alpha_k^\top - \lambda \omega_k P_k^{-1}. \tag{4.37}
$$

Note that the identity $P_k P_k^{-1} = I$, we can obtain that the update law for P_k as

$$\dot{P}_k = \omega_k \lambda P_k - P_k \omega_k \alpha_k \alpha_k^\top P_k. \tag{4.38}$$

Finally, taking the time derivative on both sides of (4.35), we solve for $\dot{\hat{\theta}}_k$ as

$$\dot{\hat{\theta}}_k = P_k \left[\omega_k \alpha_k (y - \alpha_k^\top \hat{\theta}_k) \right]. \tag{4.39}$$

We choose the control law

$$u = \begin{cases} -\dfrac{1}{2} \sum_k \bar{\omega}_k g^\top \left[\dfrac{\partial \Phi_k}{\partial x} \right]^\top \hat{\theta}_k, & \text{if } x \in \mathcal{D}^n \\ g^{-1} \left[-ke - \sum_{s=1}^{n-1} l_s x_{s+1} - \xi \text{sign}(e) \right], & \text{if } x \notin \mathcal{D}^n \end{cases} \tag{4.40}$$

The results are summarized as follows.

Control Algorithm 4.2: For system (4.1), an approximately optimal control law is (4.40) with adaptive law (4.38)-(4.39).

Remark 4.4.1 *In Control Algorithm 4.2, for $x \in \mathcal{D}^n$, the controller is adaptive for each local region. Since the approximators are localized, the number of updated parameters is smaller than that in Control Algorithm 4.2. The parameter update in one local region does not affect the values of the parameters in the other regions because the learning algorithms are localized. Therefore, the computation burden is reduced. If the size of each local region is small, the inherent approximation error can be made smaller than that in Section 4.2. So, the performance of the closed-loop system with the control law (4.40) will be better than that of the closed-loop system with the control law (4.23).*

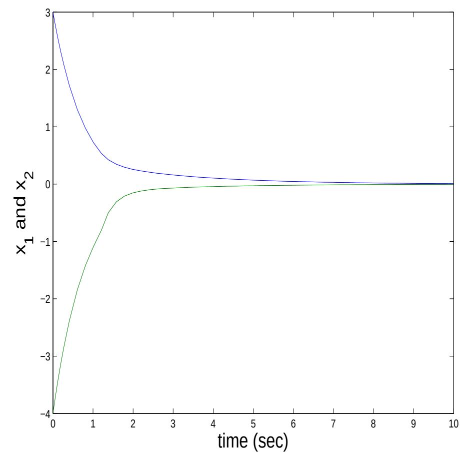

Figure 4.2: Response of x with controller (4.23)

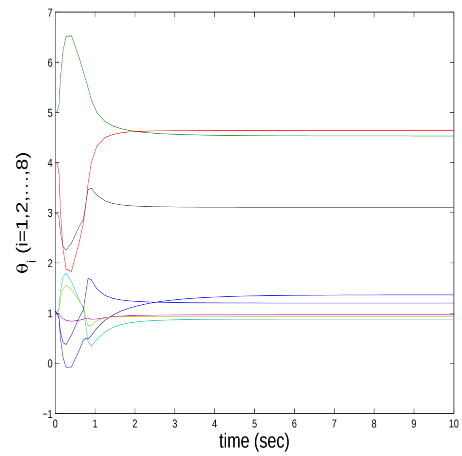

Figure 4.3: Response of $\hat{\theta}_i$ with controller (4.23)

4.5 Numerical Example

We consider for illustrative purpose a second order system by

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_1 + x_2 + x_1^3 + u.\end{aligned}$$

This system is unstable without control. For simplicity, we choose $\mathcal{D}^2 = \{x \in R^2 | \; \|x\| \leq 3\}$. For $x \in \mathcal{D}^2$, we choose $\Phi(x) = [x_1^2, x_1x_2, x_2^2, x_1^4, x_1x_2^3, x_1^2x_2^2, x_1^3x_2, x_2^4]$ and $\theta = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5,\ \theta_6,\ \theta_7,\ \theta_8]^\top$. Applying Control Algorithm 4.1, the system state converges to the neighborhood of the origin. Fig. 4.2 shows the response of the state x. Fig. 4.3 shows the response of θ_i. It can be seen that the state of the closed-loop system converges to a neighborhood of the origin and $\hat{\theta}_i$ converges to a constant.

For locally weighted learning controller (4.40), we have the similar simulation results.

4.6 Conclusion

This chapter considered the optimal control of uncertain nonlinear systems. By applying learning algorithms, adaptive approximately optimal controllers were proposed. The proposed adaptive controllers can update themselves according to estimates of the value functions. In this chapter, we did not consider optimal tracking control problem. How to solve the optimal tracking control problem based on the idea of approximately optimal control will be one of future research topics.

Chapter 5

Conclusions and Future Work

5.1 Summary of the Research

In this dissertation, we considered self-organizing based control of uncertain nonaffine systems and optimal control of uncertain nonlinear systems. In tracking control of nonaffine systems, a self-organizing on-line approximation based controller is proposed to achieve a prespecified tracking accuracy, without using high-gain control nor large magnitude switching. It was shown that our proposed controllers can achieve the prespecified tracking performance and require less computation during control. Compared with the existing results, the contribution of this part is that the tracking control of high-order nonaffine systems is solved with the aid of a self-organizing approximation and a low gain computationally efficient controller. For optimal control of uncertain nonlinear systems, we considered point-wise min-norm optimal control of uncertain nonlinear systems and approximately optimal control of uncertain nonlinear systems. Locally weighted learning observers (LWLOs) were proposed to estimate the unknown nonlinear systems. Based on the approximators that result from locally weighted learning observers, point-wise min-norm problems were defined. For the defined optimal problems, analytic controllers were proposed based on selected Lyapunov functions. The contribution of this part is that the optimal regulation and the optimal tracking control of uncertain nonlinear systems were solved by integrating locally learning algorithms to point-wise min-norm controllers. In approximately optimal control of uncertain nonlinear systems, we considered approximately optimal control of uncertain nonlinear systems. By combining the results in adaptive critic design and the approximately optimal control algorithms in [6, 62] adaptive approximately optimal control algorithms were proposed. In the proposed algorithms the controllers are updated according to the information of the value functions and converge to the optimal controllers. Compared with the offline optimal controller design algorithms in [6, 62], the proposed algorithms work online and for unknown systems.

5.2 Future Research Work

Following is a short list of possible future research topics.

- In self-organizing control of uncertain nonlinear systems, we assume the system is in the form of (2.1)-(2.3). If the uncertain nonlinear system is in the form

$$\begin{aligned}\dot{x}_1 &= f_1(x_1) + g_1(x_1)x_2 \\ &\vdots \\ \dot{x}_{n-1} &= f_{n-1}(x_1, \ldots, x_{n-1}) + g_{n-1}(x_1, \ldots, x_{n-1})x_n \\ \dot{x}_n &= f_n(x, u)\end{aligned}$$

how do we design a tracking control law?

- In point-wise min-norm control of uncertain nonlinear system, we choose the Lyapunov function to be $V = \hat{x}^\top P \hat{x}$. Obviously, this type of control Lyapunov function is not enough for a general uncertain nonlinear system. How to choose the control Lyapunov function is important open research topic.

- In point-wise min-norm control of uncertain nonlinear systems, if the model of the considered system is a more general nonlinear system, how to solve the control problem?

- In approximately optimal control of uncertain nonlinear systems, if the model is more general than that considered in Chapter 4 how can the optimal control problem be solved?

- In approximately optimal control of uncertain nonlinear systems, we do not consider the optimal tracking control problem. Can the proposed method be applied to solve the optimal tracking problem? If yes, how to design approximately optimal tracking controllers?

5.3 Author's Publication List

The following publications result from this research:

Journal Papers:

1. W. Dong and J. A. Farrell,"Cooperative control of multiple nonholonomic mobile agents," IEEE Transactions on Automatic Control, vol.53, no. 6, pp.1434-1448, 2008.
2. W. Dong and J. A. Farrell, "Decentralized cooperative control of multiple nonholonomic dynamic systems with uncertainty," Automatica, vol. 45, no. 3, pp.706-710, 2009.
3. W. Dong and J. A. Farrell, "Formation control of multiple underactuated surface vessels," IET Control Theory and Applications, vol.2, no. 12, pp.1077-1085, 2008.
4. J. A. Farrell, M. M. Polycarpou, M. Sharma, and W. Dong, "Command filtered backstepping," IEEE Transactions on Automatic Control, in press, 2008.

Conference Papers:

1. W. Dong and J. A. Farrell, "Optimal tracking control for unknown nonlinear systems based on locally weighted learning," Proc. of IEEE Conference on Decision and Control, pp.2148-2153, 2008.
2. W. Dong and J. A. Farrell, "Consensus of multiple nonholonomic systems," Proc. of IEEE Conference on Decision and Control, pp.2270-2275, 2008.
3. W. Dong and J. A. Farrell, "Formation control of multiple mobile robots with uncertainty," Proc. of American Control Conference, pp.1412-1417, 2008.

4. W. Dong and J. A. Farrell, “Optimal regulation of unknown nonlinear systems based on locally weighted learning”, Proc. of IEEE International Symposium on Intelligent Control (ISIC), pp. 1079-1084, 2008.

5. W. Dong and J. A. Farrell, “Tracking control of nonaffine systems: a self-organizing approximation approach,” Proc. of American Control Conference, pp.69-74, 2008.

6. J. A. Farrell, M. M. Polycarpou, M. Sharma, and W. Dong, “Command filtered backstepping,” Proc. of American Control Conference, pp.1923-1928, 2008.

7. V. Djapic, J. A. Farrell, and W. Dong, “Land vehicle control using command filtered backstepping approach,” Proc. of American Control Conference, pp.2461-2466, 2008.

8. W. Dong and J. A. Farrell, “Consensus of multiple uncertain mechanical systems and its application in cooperative control of mobile robots,” Proc. of IEEE Conference on Robotics and Automation, pp.1954-1959, 2008.

9. W. Dong and J. A. Farrell, “Decentralized cooperative control of multiple nonholonomic systems,” Proc. of IEEE Conference on Decision and Control, pp.1486-1491, 2007.

10. W. Dong, Y. Guo, and J. A. Farrell, “Formation control of nonholonomic mobile robots,” Proc. of American Control Conference, pp. 5602-5608, 2006.

Bibliography

[1] M. Abu-Khalaf, J. Huang, and F. L. Lewis. *Nonlinear $\mathcal{H}_2/\mathcal{H}_\infty$ Constrained Feedback Control: A Practical Design Approach Using Neural Networks.* Springer Verlag, 2006.

[2] R. C. Arkin. Cooperation without communication: multiagent schemabased robot navigation. *J. of Robotic Systems*, 9:351–364, 1992.

[3] M. Athans. The role and use of the stochastic linear-quadratic-gaussian problem in control system design. *IEEE Transactions on Automatic Control*, 16(6):529–552, 1971.

[4] C. G. Atkeson, A. W. Moore, and S. Schaal. Locally weighted learning for control. *Artifical Intelligence Review*, 11:75–113, 1996.

[5] T. Balch and R. C. Arkin. Behavior-based formation control for multirobot teams. *IEEE Trans. on Robotics and Automation*, 14(6):926–939, 1998.

[6] R. Beard, G. Saridis, and J. Wen. Galerkin approximations of the generalized Hamilton-Jacobi-Bellman equation. *Automatica*, 33(12):2159–2177, 1997.

[7] R. W. Beard, J. Lawton, and F. Y. Hadaegh. A coordination architecture for spacecraft formation control. *IEEE Trans. on Control Systems Technology*, 9(6):777–790, 2001.

[8] R. Bellman. The theory of dynamic programming. *Proc of Bat. Acad. Sci. USA*, 38, 1952.

[9] A. J. Calise, N. Hovakimyan, and M. Idan. Adaptive output feedback control of non-linear systems using neural networks. *Automatica*, 37(8):1201–1211, 2001.

[10] M. Cannon and J.-J. Slotine. Space-frequency localized basis function networks for nonlinear system estimation and control. *Neurocomputing*, 9:293–342, 1995.

[11] R. J. Caudill and W. L. Garrard. Vehicle-follower longitudinal control for automated transit vehicles. *J. of Dynamic Systems, Measurement, and Control*, pages 241–248, 1977.

[12] F.-C. Chen and H. K. Khalil. Adaptive control of a class of nonlinear discrete-time systems using neural networks. *IEEE Trans. on Automatic Control*, 40(5):791–801, 1995.

[13] J. Y. Choi and J. A. Farrell. Nonlinear adaptive control using networks of piecewise linear approximators. *IEEE Trans. on Neural Networks*, 11(2):390–401, 2000.

[14] G. Cybenko. Approximation by superposition of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, 2(4):303–314, 1989.

[15] A. J. Van der Schaft. On a state-space approach to nonlinear H_∞ control. *System and Control Letters*, 16:1–8, 1991.

[16] J. P. Desai, J. P. Ostrowski, and V. Kumar. Modeling and control of formations of nonholonomic mobile robots. *IEEE Trans. Robot. Automat.*, 17:905–908, 2001.

[17] W. Dong and J. A. Farrell. Decentralized cooperative control of multiple nonholonomic systems. *Proc. of IEEE Conference on Decision and Control*, pages 1486–1491, 2007.

[18] W. Dong and J. A. Farrell. Consensus of multiple nonholonomic systems. *Proc. of IEEE Conference on Decision and Control*, pages 2270–2275, 2008.

[19] W. Dong and J. A. Farrell. Consensus of multiple uncertain mechanical systems and its application in cooperative control of mobile robots. *Proc. of IEEE Conference on Robotics and Automation*, pages 1954–1959, 2008.

[20] W. Dong and J. A. Farrell. Cooperative control of multiple nonholonomic mobile agents. *IEEE Transactions on Automatic Control*, 53:1434–1448, 2008.

[21] J. A. Farrell and Y. Zhao. Self-organizing approximation based control. *Proc. of American Control Conference*, pages 3378–3384, 2006.

[22] J.A. Farrell and M.M. Polycarpou. *Adaptive Approximation Based Control: Unifying Neural, Fuzzy and Traditional Adaptive Approximation Approaches.* John Wiley, 2006.

[23] J. A. Fax and R. M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Trans. on Auto. Contr.*, 49:1465–1476, 2004.

[24] R. Freeman and P. V. Kokotovic. Inverse optimality in robust stabilization. *SIAM J. Control and Optimization*, 34:1365–1391, 1996.

[25] S. S. Ge and J. Zhang. Neural-network control of nonaffine nonlinear system with zero dynamics by state and output feedback. *IEEE Trans. on Neural Networks*, 14(4):900–918, 2003.

[26] S.S. Ge, C.C. Hang, T.H. Lee, and T. Zhang. *Stable Adaptive Neural Network Control.* Springer, 2001.

[27] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

[28] A. Isidori and A. Astolfi. Nonlinear H_∞ control via measurement feedback. *Journal of Mathmatics, System, Estimation, and Control*, 2:31–34, 1992.

[29] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans. on Automatic Control*, 48:988–1001, 2003.

[30] J.-P. Jiang and L. Praly. Design of robust adaptive controllers for nonlinear systems with dynamic uncertainties. *Automatica*, 34(7):825–840, 1998.

[31] R. T. Jonathan, J. Lawton, R. W. Beard, and B. J. Young. A decentralized approach to formation maneuvers. *IEEE Trans. on Robotics and Automation*, 19:933–941, 2003.

[32] H. K. Khalil. Nonlinear systems. 2nd edition. 1996.

[33] M. Krstic, I. Kanellakopoulos, and P. Kokotovic. *Nonlinear and Adaptive Control Design*. Wiley, 1995.

[34] G. Lafferriere, A. Williams, J. Caughman, and J. J. P. Veermany. Decentralized control of vehicle formations. *Systems and Control Letters*, 53:899–910, 2005.

[35] E. Lavretsky and N. Hovakimyann. Adaptive dynamic inversion for nonaffine-in-control systems via time-scale separation: part ii. *Proceedings of America Control Conference*, pages 3548–553, 2005.

[36] J. Lawton, R. W. Beard, and B. Young. A decentralized approach to formation maneuvers. *IEEE Trans. on Robotics and Automation*, 19(6):933–941, 2003.

[37] N. E. Leonard and E. Fiorelli. Virtual leaders, artificial potentials and coordinated control of groups. *Proc. of the IEEE Conf. on Decision and Control*, pages 2968–2973, 2001.

[38] F. L. Lewis, K. Liu, and Yesildirek A. Neual net robot control with guaranteed tracking performance. *IEEE Trans. on Neural Networks*, 6(3):703–715, 1995.

[39] F. L. Lewis, A. Yesildirek, and K. Liu. Multilayer neural-net robot controller with guaranteed tracking performance. *IEEE Trans. on Neural Networks*, 7(2):388–399, 1996.

[40] Z. Lin, B. Francis, and M. Maggiore. State agreement for coupled nonlinear systems with time-varying interaction. *SIAM J. of Control and Optimization*, 46(1):288–307, 2007.

[41] R. Marino and P. Tomei. *Nonlinear Control Design*. Prentice Hall International, UK, 1995.

[42] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36:789–814, 2000.

[43] L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Trans. on Auto. Contr.*, 50(2):169–182, 2005.

[44] J. Nakanishi, J. A. Farrell, and S. Schaal. Composite adaptive control with locally weighted statistical learning. *Neural Networks*, 18(1):71–90, 2005.

[45] J. Nakanishi, J. A. Farrell, and S. Schall. A locally weighted learning composite adaptive controller with structure adaptation. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 882–889, 2002.

[46] P. Ogren, E. Fiorelli, and N. E. Leonard. Cooperative control of mobile sensor networks: adaptive gradient climbing in a distributed environment. *IEEE Trans. on Auto. Contr.*, 40(8):1292–1302, 2004.

[47] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Trans. on Auto. Contr.*, 49:101–115, 2004.

[48] J. Park and I. W. Sandberg. Universal approximation using radial basis function networks. *Neural Computation*, 3(2):246–257, 1991.

[49] J.-H. Park, S.-H. Huh, S.-H. Kim, S.-J. Seo, and G.-T. Park. Direct adaptive controller for nonaffine nonlinear systems using self-structuring neural networks. *IEEE Trans. on Neural Networks*, 16(2):414–422, 2005.

[50] M. Polycarpou. Stable adaptive neural control scheme for nonlinear systems. *IEEE Transactions on Automatic Control*, 41(3):447–451, 1996.

[51] M. Polycarpou and M. Mears. Stable adaptive tracking of uncertian systems using nonlinearly parameterized on-line approximators. *International Journal of Control*, 70(3):363–384, 1998.

[52] M.M. Polycarpou. Stable adaptive neural control scheme for nonlinear systems. *IEEE Trans. on Automatic Control*, 41(3):447–451, 1996.

[53] L. S. Pontryagin. Optimal control processes. *Uspehi Mat. Nauk (in Russian)*, 14:3–20, 1959.

[54] W. B. Powell. *Approximate Dynamic Programming: Solving the Curses of Dimensionality.* Wiley & Sons Inc.: Hobroken, New Jersey, 2007.

[55] J. A. Primbs. Nonlinear optimal control: A receding horizon approach. *PhD thesis, California Institute of Technology, Pasadena, CA*, 1998.

[56] J. A. Primbs, V. Nevistic, and J. C. Doyle. Nonlinear optimal control: A control Lyapunov function and receding horizon perspective. *Asian Journal of Control*, 1(1):14–24, 1999.

[57] D. V. Prokhorov and D. C. II Wunsch. Adaptive critic designs. *IEEE Transactions on Neural Networks*, 8(5):997–1007, 1997.

[58] W. Ren and R. W. Beard. Formation feedback control for multiple spacecraft via virtual structures. *IEE Proceedings - Control Theory and Applications*, 151(3):357–368, 2004.

[59] W. Ren and R. W. Beard. Consensus seeking in multi-agent systems under dynamically changing interaction topologies. *IEEE Trans. on Automatic Control*, 50(5):655–661, 2005.

[60] R. M. Sanner and J. J. Slotine. Gaussian networks for direct adaptive control. *IEEE Trans. on Neural Networks*, 3:837–863, 1992.

[61] R. M. Sanner and J. J. Slotine. Gaussian networks for direct adaptive control. *IEEE Trans. on Neural Networks*, 3:837–863, 1992.

[62] G. N. Saridis and C.-S. Lee. An approximation theory of optimal control for trainable manipulators. *IEEE Transactions on systems, Man, and Cybernetics*, 9(3):152–158, 1979.

[63] S. Sastry. *Nonlinear Systems: Analysis, Stability, and Control*. Springer, 1999.

[64] S. Schaal and C. G. Atkeson. Constructive incremental learning from only local information. *Neural Computation*, 10(8):2047–2084, 1998.

[65] S. Shladover. Review of the state of development of advanced vehicle control systems. *Int J Vehicle Syst Dyn*, 24:551–595, 1995.

[66] S. E. Shladover. Longitudinal control of automated guideway transit vehicles within platoons. *ASME J. of Dynamic Systems, Measurement, and Control*, 100:302–310, 1978.

[67] J. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, 1991.

[68] J. J. Slotine. Sliding controller design for nonlinear systems. *International Journal of Control*, 40:421, 1984.

[69] E. D. Sontag. A Lyapunov-like characterization of asymptotic controllability. *SIAM J. Control and Optimization*, 21(3):462–471, 1983.

[70] E. D. Sontag. A 'universal' construction of Artstein's theorem on nonlinear stabilization. *System and Control Letters*, 13(2):117–123, 1989.

[71] H. G. Tanner, A. Jadbabaie, and G. J. Pappas. Flocking in fixed and switching networks. *IEEE Trans. on Automatic Control*, 52(5):863–868, 2007.

[72] H. G. Tanner, G. J. Pappas, and V. Kumar. Leader-to-formation stability. *IEEE Trans. on Robotics and Automation*, 20:443–455, 2004.

[73] D. Vrabie and F.L. Lewis. Adaptive optimal control algorithm for continuous-time nonlinear systems based on policy iteration. *Proc. of the 47th IEEE conf. on Decision and Control*, pages 73–79, 2008.

[74] D. Vrabie, O. Pastravanub, M. Abu-Khalafc, and F.L. Lewis. Adaptive optimal control for continuous-time linear systems based on policy iteration. *Automatica*, pages 477–484, 2009.

[75] Y. Zhao and J. Farrell. Self-organizing approximation for controllable nonaffine systems. *Proceedings of the IEEE Decision and Control Conference*, pages 1281–1287, 2008.

[76] Y. Zhao and J. A. Farrell. Performance-based self-organizing approximation for scalar state estimation and control. *Proc. of American Control Conference*, 2007.

[77] Y. Zhao and J. A. Farrell. Self-organizing approximation-based control for higher order systems. *IEEE Transactions on Neural Networks*, 18(4):1220–1231, 2007.